EPFL

Thèse n° 9484

Hyperdimensional computing for biosignal monitoring: Applications for epilepsy detection

Présentée le 29 juin 2023

Faculté des sciences et techniques de l'ingénieur
Laboratoire des systèmes embarqués
Programme doctoral en génie électrique

pour l'obtention du grade de Docteur ès Sciences

par

Una PALE

Acceptée sur proposition du jury

Prof. P. Frossard, président du jury
Prof. D. Atienza Alonso, directeur de thèse
Dr A. Rahimi, rapporteur
Dr T. Proix , rapporteur
Prof. J.-M. Vesin, rapporteur

École polytechnique fédérale de Lausanne

2023

Don't cry because it is over.
Smile because it happened.
— Dr. Seuss

To my grandpa Đuro . . .

For you, a thousand times over!
— The Kite Runner, Khalid Hosseini

Acknowledgements

Someone once told me that acknowledgments are the most read part of the PhD thesis (thank you, Tomás!). And I wondered why? What is in it that makes people interested in reading it so eagerly? Then I realized it is the same as the feeling of looking through windows without curtains and observing people living their life. I love to know, observe and read about people's lives, their paths, motivations, and challenges. And acknowledgments are just that about their journey called 'Ph.D..' So, let's open the window curtains of my Ph.D. path: *'What and who got me here to Switzerland to do Ph.D. on Hyperdimensional computing for epilepsy?'*

Let's start from the beginning.

I would not have decided to study engineering if there wasn't for my grandpa **Đuro** and my father, **Predrag Pale**. My childhood memories of changing car tires, fixing various devices, and brainstorming how to improvise are some of my fondest memories, and without the two of them, I would have never developed the need for understanding how things work, the feeling of power when I manage to fix something or the strong feeling of independence I have now, living in a different country. When I moved to Switzerland, the present I got from my dad was a set of tools, and they served me so many times! I'm still surprised when I find out someone doesn't have a set of screwdrivers, scalpel, duct tape, or a multimeter in their house. But maybe they just didn't have a dad like mine :). To be honest, I'm prouder of myself when I manage to improvise and fix something than now defending my Ph.D. ;)

Why do I say that? Well, going and finishing my Ph.D. was something that was just normal in the environment I grew up. From the end of primary school, when I met many of the brightest minds Croatia had, in state competitions in math and physics, I was constantly surrounded by people who were just going forward, excelling in everything they did, and keeping me also moving forward. Through them, I realized I was not weird for not being a typical girl, for being a nerd, for staying nights in high school doing experiments in physics, or for spending summers in science camps. Meeting **Jeja, Nives**, and **Barbara** at the Summer School of Science in Višnjan when I was 15 made science feel so much closer. Then, having our physics group in high school with **Toni, Antonio, Katja**, and **Juraj** was what made me feel like this is where I belong. I have to thank also everyone in **Istraživački Centar Mladih** (Youth research center),

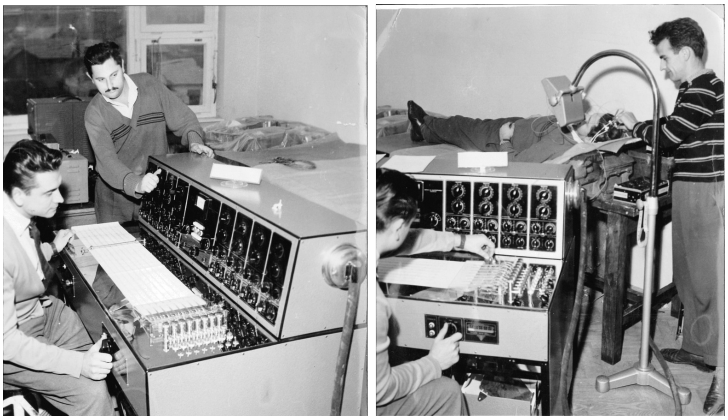

Figure 1: My grandpa Đuro Pale working on one of first EEGs in 1950s in Croatia.

which was a place for us science nerds; that just was my life for most of the high school while we were preparing for International Young Physicists tournament competitions.

However, again if there wasn't for my dad, I would have gone to study physics. And I'm grateful to him because he dared to question me, talk to me and ask me, *'How do I imagine my ideal job and one day in life in 10-20 years?'*. This helped me realize how much I cared about the application, hands-on work, and flexibility to apply my knowledge to a broad range of fields. So, I decided to study at FER (Faculty of electrical engineering and computing in Zagreb), even if I attended main physics courses on another faculty for almost two years (I was too stubborn to make the switch so easily XD). And FER was my life for another 5 years. Without **Goran, Mars**, and **Borna** (and many others), I would not have enjoyed it so much! BEST engineering competitions, Elektrijade, and ICM, made me really enjoy learning more, being an engineer and scientist. And when every one of the previously mentioned people went to do their Ph.D. in Canada, Austria, Switzerland, England, or Croatia, there was little doubt that it was something for me too.

Why, then, Switzerland? Well, I didn't want to leave Croatia, but I was too annoyed by people saying how things are so much better elsewhere. And I wanted to understand what it is that is so much better there. Can Croatia be as good as Switzerland? Can FER be as good as ETHZ or EPFL? Instead of playing broken telephone, and I needed to know firsthand! And there were mountains! Mountains that are now my place for my soul. When I cannot be in my grandma's garden, at least I can go to the mountains.

So I sent applications to EPFL, and after a bit of waiting, I joined UP Hummel lab. Ending up in a neuroscience lab was the result of me falling in love with biomedical engineering on my Erasmus exchange in TU Vienna with **Goran**, during my master thesis under supervision of **Prof. Mario Cifrek**, and stories from my grandpa (going back to you, grandpa!) about making the first electroencephalographs in Zagreb in 1950's(look at the picture above :))! I have to admit being an engineer in a lab of doctors, neuroscientists, and neuropsychologists was not the easiest. But it was my home for almost two years, and it was where I met my dearest friends in Switzerland. It was a challenging time, but it was also a time that made me aware of who I

am, what I care about, and what I need to work on. **Andeol, Claudia, Elena, Julia**, and **Pablo**, my Bikers :) (I ordered you by alphabet because I didn't know how to start XD), I don't know how I would survive those first two years without you!

And then came a change, sudden, but one of those things when life closes to you one door, another one opens. I moved to the ESL lab of **Prof. David Atienza Alonso**. Thank you, **Matteo** and **Greg**, for convincing me that a lab, where the biggest problem is that they don't let you grow tomatoes in your office, was a good option :P. People who deserve my greatest gratitude were **Dr. Adriana Arza Valdés** and **Dr. Tomás Teijeiro Campo**, who worked to navigate me through my Ph.D. Tomas was there from my first interview to the end of my Ph.D., and he was the one who introduced me to the topic of Hyperdimensional computing, which ended up being a central part of my Ph.D. Thank you for all the meetings, last moment paper corrections, ideas, and support! This Ph.D. would have been very different and would probably last much longer without you.
Finally, all this would not be possible if **David** didn't decide to give me a chance. David, thank you for everything! Thank you for opening my eyes to how one can do amazing science while having a family, being physically active, and at the same time knowing what each of the >40 people in the lab is doing and how to help them. This was really inspiring! Thank you for everything I learned in the lab, for all the experience, for allowing me to take courses (even if I took too many of them), and for gaining experience with conferences (despite COVID). Thank you for the trust and understanding you had and for giving me lots of freedom in research! It might have been good, or even not so good, probably you would have done it better, but I felt that it was my own!
I want to also thank all the members of the **PEDESITE project** on which I worked. I enjoyed working with all of you! I enjoyed the experience of working on a big multi-center project with three universities and hospitals involved. Collaboration between academia and the ultimate users of 'our scientific results' was such an important experience!

Similarly, time in UP Hummel also made me realize that science is a bit like 'a playground' for adults, and it made me feel like doing only science can be selfish. I started searching for other options, and here I'm thankful for the Effective Altruism community of EPFL and Switzerland, EPFL's summer school on open science, and the endless additional courses from EPFL I attended. All of this will play a role in my future, where I hope to stay connected to science and work on improving science and how science can influence the world but probably not from the inside.
Thank you to everyone in **Penkala** (Association of young Croatian scientists), my side passion and where I spend most of my free time. Organizing Mutimir, starting the first Croatian science podcast, writing blogs with Croatian scientists, raising money for projects, and so much more made me realize how much making (even if small) changes energizes me. It also allowed me to stay close to Croatia even if I'm physically far.

And finally, back to the ESL lab. Thank you, **Lara, Renato, Benoit, Eli, Tim, Greg**, and **Fabio**,

for making me feel accepted when I joined the lab. Thank you for all the hikes, climbs, and chill time! **Lara**, thank you for being my connection to Croatia, even in the lab. It meant a lot to be able to just chat during lunchtime. **Renato**, thank you for being the person who, when called the evening before, you were ready for any outdoor action the next day. Endless hikes, skiing, ferratas, from COVID till the day you left :(. Thank you for being our 'third wheel'!

'Third wheel to what?' To **Andrew**! My, now already, husband, who I also met in the ESL lab. His flannel shirts, adventurous mindset, kindness, goofiness, and endless support. Thank you, 'first doors for getting closed'! With you life is so much more; more fun, more adventures, more deep discussions, more projects, more fun, more everything! Thank you for being you! Volim te Bubi!

Back to **my family: mama, tata, Mina, bake i dede, Tona, Lina, Sanja**, hvala vam za svu podršku! Što ste uvijek samnom, što trpite što me nema da pomognem oko svega, što kad dođem uvijek imate vremena za mene, što trpite moje telefonske pozive iz auta ili dok radim 100 drugih stvari u paraleli. Volim vas najviše na svijetu!

I want to thank Lausanne and Switzerland for being such an incredible place. A place where I can go skiing and swimming on the same day. A place where my tables with ideas for trips, hikes, and climbs never get exhausted. A place where I grew into an independent person ready for anything that life throws at me :). Thank you Lausanne!
EPFL, thank you for being such an amazing place and for all the great things I learned from how you work and how you are structured. Hopefully, one day I will be able to apply all that in Croatia.

Last but not least, I need to thank also my jury members **Dr. Abbas Rahimi, Dr. Timothée Proix, Prof. Jean-Marc Vesin** and **Prof. Pascal Frossard** for reading all 200 pages of this thesis, the great discussions, and the ideas for improving my thesis. I really enjoyed my defense!

To conclude, I have mixed feelings about the end of my Ph.D. I will miss being a student, I will miss research, collaboration, and supervising students. But I'm looking forward to new doors being opened and the challenges (and joy) they will bring!

Lausanne, 30 May 2023 U. P.

Abstract

Hyperdimensional (HD) computing is a novel approach to machine learning inspired by neuroscience, which uses vectors in a hyper-dimensional space to represent data and models. This approach has gained significant interest in recent years with applications in various domains. Its advantages, such as fast and energy-efficient learning and the potential for online and privacy-preserving distributed learning, make it interesting for power-efficient applications, such as continuous biosignal monitoring via wearable devices. However, wearable biomedical applications pose a broad range of challenges for hyperdimensional computing that must be tackled before its potentially widespread adoption, such as encoding the spatio-temporal data to hyperdimensional vectors and learning from highly unbalanced and large datasets.

I focus on epilepsy detection as a use case and inspiration for further improving and testing the proposed HD computing methods. Epilepsy is a chronic neurological disorder that affects a significant portion (0.6 to 0.8%) of the human population and imposes severe risks in the daily life of patients. One-third of patients still suffer from seizures despite pharmacological treatments, demonstrating a need for solutions that allow continuous unobstructed monitoring and reliable detection (and ideally prediction) of seizures. However, despite advances in machine learning and Internet of Things (IoT), small and non-stigmatizing wearable devices for continuous monitoring and detection in outpatient environments are not yet widely available, and the few existing commercial ones are focused on limited and very specific types of seizures.

In this thesis, I demonstrate and develop additional aspects in which HD computing, and the way its models are built and stored, can be used to understand further, compare, and create more complex machine-learning models for epilepsy detection. These possibilities are not feasible with other state-of-the-art models, such as random forests or neural networks.

More specifically, I propose new approaches to improve two main parts of the HD computing workflow: encoding and learning. Different methods to encode three-dimensional sets of information (features with spatial and temporal information) are proposed and discussed. Next, due to the highly personalized nature of epileptic seizures and their unbalanced nature, learning is improved by proposing a new multi-centroid learning approach.

Then, how HD computing can be used to build personalized and generalized models is presented. More specifically, I study the process of creating generalized models from personalized ones for future distributed learning applications, and investigate the evolution of generalized models as more individual models are added. HD computing enables combining personalized and generalized models forming hybrid models, resulting in increasingly performant epilepsy detection. Finally, HD computing is utilized to test the knowledge transfer between models created on two different datasets, making a first step towards the integration of knowledge from available epilepsy datasets.

As large and well-annotated datasets are still instrumental for building models that perform well on unseen data, I identify a wide range of methodological decisions that must be made and reported when training and evaluating the performance of epilepsy detection systems using unbalanced, long-term recordings. In particular, I characterize the influence of individual choices, providing a broader picture of each decision and giving, where possible, good-practice recommendations based on our experience.

The need for interpretable models and predictions in healthcare applications is paramount. Thus, this PhD thesis demonstrates the possibility of HD computing for visualizing prediction decisions in time, per features, and also per channels. Also the process of feature and channel selection using HD computing encoding is explored.

In the end, I led the development of HDTorch, an open-source PyTorch-based library, further extended with custom CUDA-backed hypervector operations. HDTorch enables much faster exploration and development of HD computing algorithms.

Overall, through this thesis, I demonstrate how hyperdimensional computing can help bring wearable and interpretable healthcare systems closer to reality and patients' everyday life. Epilepsy detection is used as a representative use case, as it is challenging not only from a medical understanding and socio-economic aspects but also from an engineering perspective. However, all the work proposed is easily translatable to other biomedical signals and applications. Thus, I believe this work can inspire and foster further improvements in hyperdimensional computing and its applications in wearable healthcare applications.

Key words: Hyperdimensional computing, Biosignal monitoring, Wearable healthcare applications, Epilepsy, Epileptic seizures, Interpretable machine learning, Large unbalanced dataset, Personalized and generalized model, Knowledge transfer

Résumé

L'informatique hyperdimensionnelle (HD) est une nouvelle approche de l'apprentissage automatique inspirée des neurosciences, qui utilise des vecteurs dans un espace hyperdimensionnel pour représenter des données et des modèles. Cette approche a gagné en intérêt ces dernières années avec des applications dans divers domaines. Ses avantages, tels que l'apprentissage rapide et économe en énergie et son potentiel pour d'apprentissage en ligne et distribué en préservant la confidentialité, le rendent intéressant pour des applications économes en énergie, telles que la surveillance continue des biosignaux via des appareils portables. Cependant, les applications biomédicales portables posent un large éventail de défis pour l'informatique hyperdimensionnelle qui doivent être relevés avant son adoption potentiellement généralisée, tels que l'encodage des données spatio-temporelles en vecteurs hyperdimensionnels et l'apprentissage à partir d'ensembles de données très déséquilibrés et volumineux.

Je me concentre sur la détection de l'épilepsie en tant que cas pratique et source d'inspiration pour améliorer et tester davantage les méthodes de calcul HD proposées. L'épilepsie est un trouble neurologique chronique qui affecte une partie importante (0,6 à 0,8%) de la population humaine et engendre de graves risques dans la vie quotidienne des patients. Un tiers des patients souffrent encore de crises malgré les traitements pharmacologiques, démontrant un besoin de solutions permettant une surveillance continue sans entrave et une détection fiable (et idéalement une prédiction) des crises. Cependant, malgré les progrès de l'apprentissage automatique et de l' Internet des objets (IdO), il existe encore un manque de dispositifs de surveillance et de détection continues en environnements ambulatoires qui sont léger, portables et non stigmatisant. Les quelques dispositifs commerciaux existants se concentrent sur des types de crise limités et très spécifiques.

Dans cette thèse, je démontre et développe des aspects supplémentaires dans lesquels l'informatique HD et la façon dont ses modèles sont construits et stockés, peuvent être utilisés pour mieux comprendre, comparer et créer des modèles d'apprentissage automatique plus complexes pour la détection de l'épilepsie. Ces possibilités ne sont pas réalisables avec d'autres modèles de pointe, tels que les forêts aléatoires ou les réseaux de neurones.

Plus spécifiquement, je propose de nouvelles approches pour améliorer deux composantes principales du workflow informatique HD : l'encodage et l'apprentissage. Différentes méthodes pour coder des ensembles d'informations tridimensionnelles (caractéristiques ayant des informations spatiales et temporelles) sont proposées et discutées. Par la suite, en raison du caractère hautement personnalisé des crises d'épilepsie et de leur nature déséquilibrée, l'apprentissage est amélioré en proposant une nouvelle approche d'apprentissage multi-centroïde.

Ensuite, comment l'informatique HD peut être utilisée pour construire des modèles personnalisés et généralisés est présentée. Plus précisément, j'étudie le processus de création de modèles généralisés à partir de modèles personnalisés pour de futures applications d'apprentissage distribué, et j'étudie l'évolution des modèles généralisés au fur et à mesure que des modèles individuels sont ajoutés. L'informatique HD permet de combiner des modèles personnalisés et généralisés pour former des modèles hybrides, résultant en une détection de l'épilepsie de plus en plus performante. Enfin, l'informatique HD est utilisée pour tester le transfert de connaissances entre les modèles créés sur deux ensembles de données différents, faisant un premier pas vers l'intégration des connaissances à partir des ensembles de données disponibles sur l'épilepsie.

De grands ensembles de données bien annotés sont toujours essentiels pour construire des modèles qui fonctionnent sur de nouvelles données. J'identifie ainsi un large éventail de décisions méthodologiques qui doivent être prises et rapportées lors de l'entrainement et de l'évaluation des performances des systèmes de détection de l'épilepsie utilisant des données déséquilibrées et desenregistrements de longue durée. En particulier, je caractérise l'influence des choix individuels, offrant une vision plus large de chaque décision et donnant, lorsque cela est possible, des recommandations de bonnes pratiques basées sur notre expérience.

Le besoin de modèles et de prédictions interprétables dans les applications médicales est primordial. Ainsi, cette thèse démontre la possibilité du calcul HD pour visualiser les décisions de prédiction dans le temps, par caractéristiques, mais aussi par canaux. Le processus de sélection des fonctionnalités et des canaux à l'aide de l'encodage informatique HD est également exploré.

En fin de compte, j'ai dirigé le développement de HDTorch, une bibliothèque open-source basée sur PyTorch, étendue avec des opérations hypervectorielles personnalisées basées sur CUDA. HDTorch permet une exploration et un développement beaucoup plus rapides des algorithmes de calcul HD.

Globalement, à travers cette thèse, je démontre comment l'informatique hyperdimensionnelle peut aider les dispositifs de santé portables et interprétables à se rapprocher de la réalité et de la vie quotidienne des patients. La détection de l'épilepsie est utilisée comme pratique, car

elle représente un défi non seulement d'un point de vue médical et socio-économique, mais également d'un point de vue technique. Cependant, tous les travaux proposés sont facilement transposables à d'autres signaux et applications biomédicales. Ainsi, je crois que ce travail peut inspirer et favoriser de nouvelles améliorations dans l'informatique hyperdimensionnelle et ses applications dans les dispositifs médicaux portables.

Mots clefs : Informatique hyperdimensionnelle, surveillance des biosignaux, dispositifs médicaux portables, épilepsie, crises d'épilepsie, apprentissage automatique interprétable, vaste ensemble de données déséquilibré, modèle personnalisé et généralisé, transfert de connaissances

Contents

List of Figures

List of Tables

Acronyms

ACM Accelerometry.
AED Antiepileptic Drug.
AI Artificial Intelligence.
BSD Binary Spatter Code.
BSDC Binary Sparse Distributed Code.
CNN Convolutional Neural Network.
DL Deep Learning.
DRE Drug Resistant Epilepsy.
DT Decision Tree.
ECG Electrocardiography.
EEG Electroencephalography.
EMD Empirical Mode Decomposition.
EMG Electromyography.
FAR False Alarm Rate.
fMRI functional Magnetic Resonance Imaging.
fNIRS functional Near Infrared Spectroscopy.
HDC Hyperdimensional Computing.
HRR Holographic Reduced Representation.
ICA Independent Component Analysis.
IEEG Intracranial EEG.
ILAE International League Against Epilepsy.
IoT Internet of Things.
KNN K-Nearest Neighbors.
LBP Local Binary Patterns.
LDA Linear Discriminant Analysis.
MAP Multiply-Add-Permute.
MEG Magnetoencephalography.
ML Machine Learning.
NN Neural Network.
NNMF Non-Negative Matrix Factorization.
PCA Principal Component Analysis.
PPG Photoplethysmography.
PPV Positive predictive value.

QoL Quality Of Life.
RB Rolling Base Cross Validation.
RF Random Forest.
RNN Recurrent Neural Network.
ST Spatio-Temporal Data.
SUDEP Sudden Unexpected Death in Epilepsy.
SVD Singular Value Decomposition.
SVM Support Vector Machine.
TCS Tonic-Clonic Seizure.
TPR True Positive Rate.
TSCV Time Series Cross Validation.
VSA Vector Symbolic Architectures.
WD Wearable Device.
WSDD Wearable Seizure Detection Device.

1 Introduction

1.1 Wearable healthcare applications

All over the world, healthcare quality is increasing, but at the same time, healthcare costs are rapidly increasing too. Increasing life expectancy, the increasing prevalence of chronic diseases, and the development of costly new therapies contribute to this trend. Thus, it comes as no surprise that scholars predict a grim future for the sustainability of healthcare systems [1]. Artificial intelligence (AI), machine learning (ML), and wearable devices (WD), however, promise to alleviate the impact of these developments by improving healthcare and making it more cost-effective [2]. More specifically, the healthcare landscape has been changing rapidly in the last years due to several factors:

- Mobile, wearable, and Internet of Things (IoT) technologies have achieved huge advancements in processing power, battery lifetime, and size, and have become a part of our everyday life. Smartphones, watches, and bracelets are equipped with miniature sensors to monitor biosignals that are useful for diagnosing various health problems.
- Signal processing and machine learning algorithms have advanced and have been applied to a broad range of medical problems, such as the detection of breast cancer [3], classification of skin cancer [4], diagnosis of Alzheimer's disease [5], diagnosis of diabetic retinopathy in retinal images [6] and many more. Interdisciplinary work and collaboration between researchers, scientists, clinicians, and medical personnel is becoming the new normal. Thus, experts in signal processing, data mining, and machine learning are actively involved in proposing solutions for various biomedical applications.
- The availability of large-scale biomedical data: Large datasets are becoming available due to advances in mobile and wearable technologies, storage devices, and tools to analyze them. Today, the fields with the largest datasets are radiology [7], cardiology [8], pathology [9], and genomics [10] where they are using ML approaches for automatic diagnostics, classification, and prediction.

Moreover, the healthcare system is also moving from reactive disease intervention to proactive prevention [11], as it is not only more cost-effective, but it usually leads to a higher Quality

of life (QoL) [12]. At the same time, healthcare is shifting from a one-size-fits-all model to personalized medicine [13] and from institution-centered to decentralized [14]. For all these paradigm shifts, novel algorithm design and optimization for lightweight and wearable IoT devices are essential [15]. These kinds of devices can be utilized in a broad range of cases, from general continuous monitoring, to early detection of diseases and preventive healthcare.

Thus, a lot of research has been done on the design of new machine learning algorithms for the high-quality monitoring and detection of various diseases. Research is directed by several requirements: high prediction/detection accuracies with low false alarm rates, interpretable results, lightweight and fast models, potential for continuous learning, and personalizing of the models.

One ML approach that has gained significant interest in recent years is HyperDimensional (HD) computing. It is an approach inspired by neuroscience, which uses vectors in a hyper-dimensional space to represent data and models. Its advantages, such as fast and energy-efficient learning and the potential for online and privacy-preserving distributed learning, make it interesting for power-efficient applications, such as continuous biosignal monitoring via wearable devices. However, wearable biomedical applications pose a broad range of challenges for hyperdimensional computing that must be tackled before its potentially widespread adoption, such as encoding the spatio-temporal data to hyperdimensional vectors and learning from highly unbalanced and large datasets.

Thus, the focus of this thesis is exploring and advancing the potential of hyperdimensional computing for biomedical signal monitoring. As a main use case, we use epilepsy detection since it is a good representative of all challenges mentioned above, as will be explained in the next chapter. Moreover, epilepsy is a challenging medical problem for which still widely spread wearable monitoring devices do not exist.

1.2 Epilepsy

1.2.1 Neurological disorder

Epilepsy is a chronic neurological disorder characterized by the unpredictable occurrence of seizures. It affects a significant portion of the world's population, with a prevalence of 0.6 to 0.8% [16], [17], making it one of the most common neurological diseases [18], in addition to migraine, stroke, and Alzheimer's disease [19]. Moreover, epilepsy represents the second neurological cause of years of potential life lost, only behind stroke [20], mainly due to accidents triggered by seizures and sudden unexpected death in epilepsy (SUDEP).

The causes of epilepsy are not yet fully understood, but the proposed classification is into idiopathic (with a presumed genetic basis), symptomatic (resulting from a structural abnormality), or cryptogenic (resulting from an unknown underlying cause) [21]. Epilepsy affects people of all ages, from newborns to the elderly. Distribution over age groups is not equal, as shown in Fig. 1.1. Epilepsy is the most common in elderly people above 65, but it also manifests quite frequently in the young and teenage years.

When discussing types of seizures, there are many types, each with its own characteristics and features. In 2017, the International League Against Epilepsy (ILAE) revised its classification of seizures to make diagnosis and classification more accurate and simpler [22]. There are three main characteristics of seizures used for classification:

- **Location of seizure onset.** Most common are focal seizures, formerly known as 'partial' seizures, that occur in 60% of the patients [22]. These start with a relatively local onset in one hemisphere and sometimes spread through the whole brain. These seizures can often be subtle or unusual and may go unnoticed. In case of a spread to the other hemisphere, a seizure is called 'focal-bilateral' (the former name being 'secondarily generalized'). Seizures can also be generalized (in 30% of the patients), with seizures starting simultaneously over the whole brain. Finally, there is a small portion (10%) of seizures for which the beginning of the seizure is not fully understood.
- **State of awareness.** During a seizure a person often loses consciousness. Awareness doesn't refer to whether the person was aware of the seizure itself. Focal seizures can be further classified as "focal aware" (previously called "simple partial seizures") or "focal unaware" (previously called "complex partial seizures"). Since generalized onset seizures almost always affect awareness in some way, no additional descriptor on awareness is used with generalized seizures.
- **Motor and nonmotor symptoms.** Seizures, primarily generalized seizures, can have a physical component, such as a fall or muscle contraction (jerks, twitching, stiffness of muscles, etc.), or they may have no physical component. The most common motor symptoms are stiffening (tonic) and jerking (clonic), resulting in the common name 'tonic-clonic' for such seizures. Seizures with no physical component are referred to as 'absence seizures' [22]. Nonmotor seizures can have other symptoms, such as the

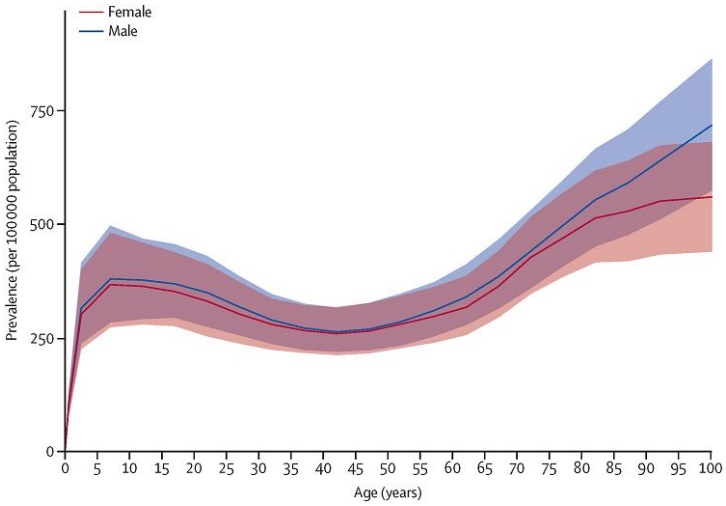

Figure 1.1: Incidence and prevalence of epilepsy across the lifespan. The picture is taken from [23])

disruption of blood pressure, heart rhythm or bladder function, cognitive impairment or tingling, or smelling an unpleasant odor.

Most people will only have one or two types of seizures, which can vary in severity. However, people with severe / complex epilepsy or significant brain damage may experience several different types of seizures [22]. Seizures can vary a lot in duration, ranging from only a few seconds to several minutes. A simple visualization of all seizure types and their subclassifications is illustrated in Fig. 1.2 [22].

Seizure control by antiepileptic drugs achieves success in 70% of patients [24], but only 15% of those patients achieve full seizure control with zero side effects [25]. Additionally, 7 to 8% of the patients can be treated by surgical operation, but the remaining 25 to 30% of patients suffer from Drug Resistant Epilepsy (DRE). DRE patients rely on various strategies for seizure monitoring, detection, and prediction to improve their lives [26].

Due to its high inter-patient variability, unpredictable nature, and not yet completely understood origins, the topics of seizure detection and treatment still pose open research questions. Furthermore, no wearable devices are yet available for the prediction, detection, or continuous monitoring of all types of seizures in outpatient settings, while there is a clear need for such solutions [27], [28]. Moreover, these solutions are instrumental in designing novel treatments, assisting patients in their daily lives, and preventing possible accidents. This need is also evident in the growing number of studies and publications on seizure detection methods [29], [30] and wearable devices [31], [32].

1.2.2 Monitoring epilepsy

To detect epilepsy and extract relevant parameters such as frequency, seizure type, or location, electroencephalography (EEG) or Intracranial EEG (iEEG) recordings are usually needed. EEG

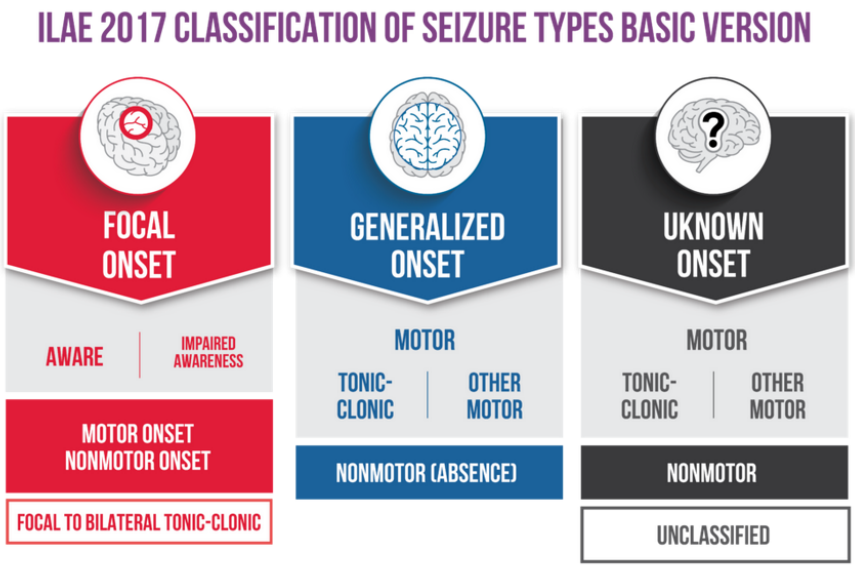

Figure 1.2: Classification of epileptic seizure types according to new ILAE classification [22]. The picture is taken from www.cureepilepsy.org.

is by far the most widely adopted clinical technique for diagnosing, detecting, anticipating, and predicting seizures in clinical practice [16], [33], [34]. An EEG measures the brain's electrical activity on the scalp surface and is non-invasive, and is thus cheaper and easier to obtain than an iEEG. It is usually recorded in clinical settings, but there are wearable devices being developed that enable more and more EEG recordings in outpatient environments, such as Byteflies's Sensor-Dot [35], eGlasses [36], and behind-the-ear [37] and in-ear devices [38]. An iEEG, on the other hand, is invasive but has a higher signal-to-noise ratio and is less affected by artifacts and environmental factors [39]. It is able to capture epileptiform discharges with much higher quality than on-scalp EEG recordings, which are highly attenuated from the source to the scalp [40].

To diagnose and characterize epilepsy, the most commonly used in clinical settings is video EEG (v-EEG), where a patient spends several days in a hospital while being tracked using camera recording, audio recordings, EEG, and often electrocardiogram (ECG) recordings. Patients are also monitored by medical personnel who record when seizures occur and visually characterize them (e.g., if a patient lost consciousness, what type of motor or non-motor symptoms they had, etc.). This is as well the typical procedure for how most of the current epilepsy databases were collected.

As a seizure is a holistic phenomenon that causes changes in the whole body, other biosignals exhibit changes during seizures as well. More specifically, the main hypothesis is that epileptic seizures are caused due to dysregulations of the Sympathetic and Parasympathetic Nervous System (SNS/PNS), so heart activity which is also modulated by the Autonomic Nervous System (ANS), provides an indicator of the ictal onset. Further, Electrodermal Activity (EDA),

also called galvanic skin response, is an autonomic marker for sympathetic skin activity which also exhibits unique properties during seizures. Finally, electromyography (EMG) is a self-evident modality for detecting seizures with motor components. EMG recordings measure the electrical activity of muscles, thus identifying ictal muscle contraction. As EEG and iEEG are the least practical modalities for unobstructed wearable outpatient monitoring, devices containing multiple other signals, such as EMC, accelerometry (AMC), photoplethysmography (PPG), EDA, and skin temperature, are expected to be the main path for wearable outpatient epilepsy monitoring.

1.2.3 The impact of epilepsy

Impact on patients

The impact of epilepsy is complex, multifaceted and far-reaching, not only for the individual living with the condition but also for their family and the wider society [41]. This is illustrated in Fig. 1.3. First, as mentioned before, the occurrence of seizures is unpredictable and often dangerous, increasing the risk of injury, hospitalization, and mortality [42]. The most frequent injuries among individuals with epilepsy are contusions, fractures and brain concussions, wounds, strains, and burns. People with epilepsy have a 5% chance per year of visiting an emergency department due to an injury resulting from a seizure [43]. Moreover, in case of a loss of consciousness, a seizure can lead other serious injuries, such as to driving accidents or drowning.

Further, many studies have shown that patients with epilepsy are at a higher risk, compared with the general population, of a wide range of medical and psychiatric comorbidities [44], [45]. Fig. 1.4, from [23] illustrates a broad range of comorbidities in older people with epilepsy, out of which many are not age specific. The US National Comorbidity Survey Replication [44], which included over 5500 individuals, showed that patients with epilepsy are significantly more likely (93.6% vs. 77.8%) to have at least one physical comorbidity and at least one mental disorder (67.9% vs. 47%) when compared to people without epilepsy. Some of the common medical comorbidities are stroke, asthma, hearing and visual impairments, headaches, and digestive disorders. It is widely accepted that comorbidities have an impact on the overall health status and QoL of patients. For epilepsy patients specifically, comorbidity has been shown to influence both QoL and also subjective health status [46], [47].

Unfortunately, for the person suffering from epilepsy, the burden is not confined simply to the physical risks of seizures and the risks of treatment. The stigma in many cultures still attached to persons with the condition often results in social exclusion and isolation, as well as difficulties in accessing education and employment options as freely as people without epilepsy [41]. Furthermore, epilepsy is highly associated with mental health and psychiatric disorders. For example, the association between depression and epilepsy is particularly well established and seems to be bidirectional [48], [49]. More specifically, although stress, anxiety, and depression were each found to be significant predictors of time since the last

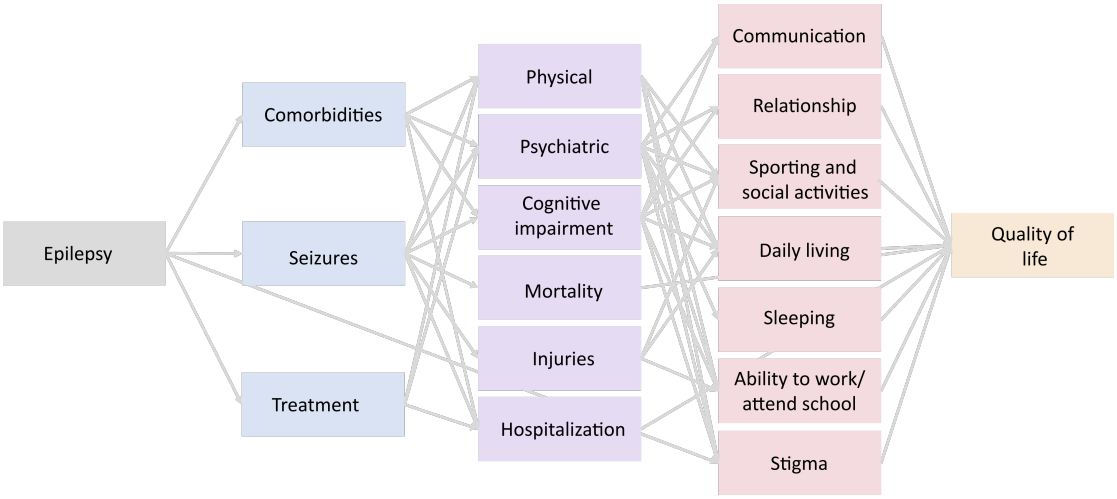

Figure 1.3: The impact of epilepsy on patients' lives. The picture is taken from [53]

seizure, depression was found to mediate the relationship between anxiety, stress, and seizure frequency [50]. Moreover, individuals with epilepsy are also more likely to report lifetime anxiety disorders or even suicidal thoughts [17], [51]. The QoL of patients is strongly influenced by the frequency and severity of the seizures [52]. The anxiety about seizures ('seizure worry') further influences the QoL of patients [46].

Beyond the effects of epilepsy itself, Antiepileptic Drug (AED) treatment is commonly associated with side effects that further impair patients' QoL. More specifically, data from randomized control trials report that up to 90% of patients with epilepsy experience AED side effects [54]. A European study [55] on over 5000 people with epilepsy found that the most common side effects of AED are tiredness (58%), memory problems (50%), concentration difficulties (48%), sleepiness (45%), difficulties in thinking clearly (40%) and nervousness and agitation (36%). Patients are often reluctant to discuss these issues with their physicians, resulting in the underestimated perception of the frequency of side effects by physicians [56]. Unfortunately, studies also show that the side effects alone account for up to 40% of treatment failures [54], [57]. In the end, the side effects of AED are often more disabling and reduce patient QoL more than the seizures themselves [58]. Interestingly, studies have found that if seizure freedom cannot be achieved, patients are more likely to benefit from reducing the burden of AED and depressive comorbidity than from the actual reduction of seizure frequency [59].

In the end, the mortality rate is 2 to 3 times higher for individuals with epilepsy than in the general population, the highest in the young demographic and between 5 and 10 years after diagnosis [60], [61]. Observations suggest that the higher mortality is partly related to the underlying disorder causing epilepsy rather than a direct consequence of the seizures. Yet, for patients with chronic epilepsy, seizure-related deaths may account for up to 40% of deaths [61]. SUDEP is one of the most directly seizure-related causes of death [62], [63].

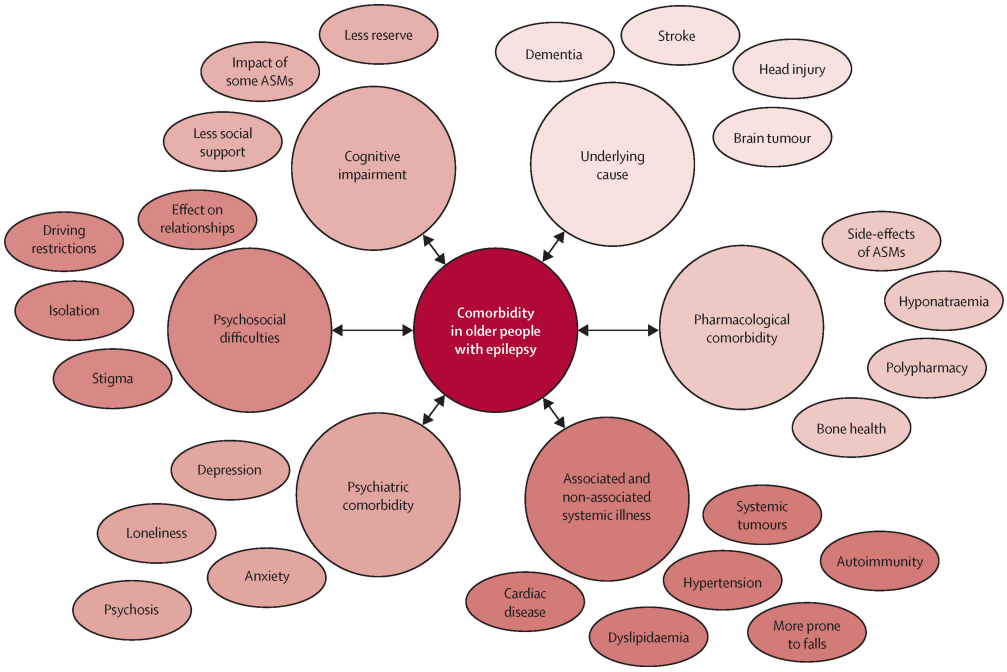

Figure 1.4: Comorbidities in older people with epilepsy. The picture is taken from [23])

Economic burden

The high level of comorbidity associated with epilepsy has an impact not only on the patients themselves, but also on the healthcare system as a whole [64], [65]. The Treatment in Geriatric Epilepsy Research study [66], on more than 800,000 elderly patients, found that the hospitalization rate for patients with epilepsy was 3 times higher compared to elderly patients without epilepsy. Moreover, approximately 80% of overall medical costs for patients with epilepsy are for non-epilepsy-related care [64].

In the United States, the economic costs of epilepsy are highly variable, but the estimates are enormous. A study in 2000 [67] estimated annual costs of $12.5 billion, 14% of which was attributed to direct costs and 86% to indirect costs. More specifically, a study [68] analyzing costs in the privately insured patient population in the US between 1999 and 2004 found that, on average, direct annual costs were significantly higher per patient with epilepsy than per control ($10,258 vs. $3,862). Data from the study [68] are consistent with a study [69] carried out on member countries of the European Union. The estimated total cost of the disease in Europe was 15.5 billion euros in 2004, again with indirect costs being the single most dominant cost category. The total cost per case was between 2,000 and 11,500 euros, as shown in Fig. 1.5.

A recent review paper [70], analyzing more than one hundred studies, estimated that epilepsy affected 52.5 million people around the world, leading to total costs of more than 119.3 Billion. Even if most of the costs are in the wealthiest countries [70] (where a minority of the world's epilepsy population reside), the economic burden of epilepsy is enormous also in developing countries. For example, in India, it is estimated to be 88.2% of Gross National Product (GNP)

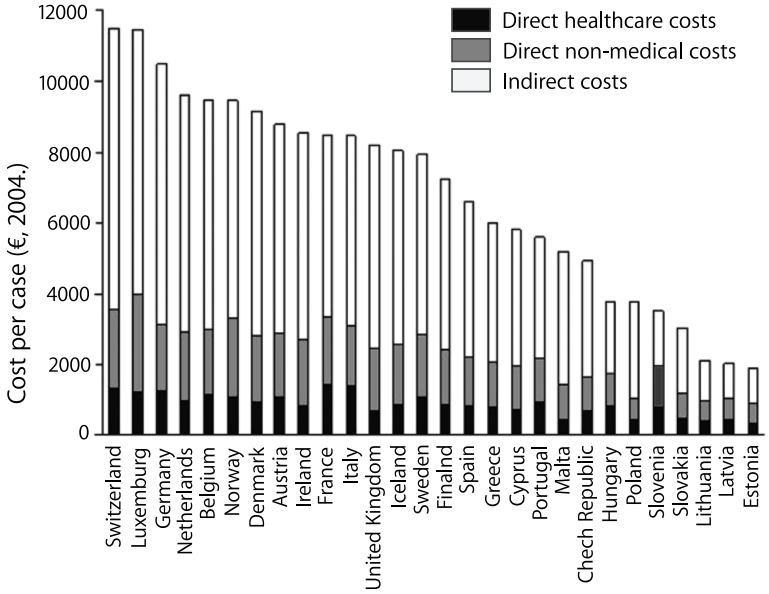

Figure 1.5: Cost per case in epilepsy in Europe in 2004. The picture is taken from [69])

per capita and 0.5% of the overall GNP [71].

The authors of a recent study [70] suggest that in high-income countries, the focus should be on improving the efficacy of health care delivery and also lowering barriers to employment for people with epilepsy. On the other hand, in lower-income countries, the relatively low cost of epilepsy reflects the lack of available healthcare services, and thus the main focus of future policies and healthcare programs should be to increase the availability of healthcare services for patients with epilepsy.

1.2.4 State of the art in algorithms

Seizure detection, prediction and localization
Automatic monitoring of seizures from biosignals has been interesting for clinicians and engineers for a while. Seizure monitoring algorithms can be divided most clearly into two categories: 1) seizure detection and 2) seizure prediction. Seizure detection means detecting a seizure as soon as it starts to happen, whereas seizure prediction tries to detect the beginning of the preictal (before-seizure) period as early as possible (minutes to hours) [72]. Seizure prediction has the potential to issue warnings to patients or caregivers and potentially even prevent seizures or injuries, but it is also a more complex task than detection. It usually assumes classifying between pre-ictal and inter-ictal patterns. The problem is that there is no global consensus on the definition of the pre-ictal period. Thus, in most works, researchers take different time values to define the start of the pre-ictal period.

Another essential step needed before epileptic surgery is localization, which means identifying a point/location/region of the brain affected by a seizure. The 10–20 positioning system gives some clues for identifying the location of a seizure, but localization is easier and can be more precise when using iEEG recordings. Recently, various computational and machine learning methods have been applied to identify a seizure location [73]–[78].

Signal modalities
EEG is the most important physiological signal for the detection of epileptic seizures, and it has been used in the largest number of studies working on automatic seizure detection/prediction. Due to a large number of channels (usually between 2 and 20 for scalp EEG and >20 and up to several hundreds for iEEG), the number of features that can be extracted is huge. Furthermore, changes in the EEG signals are almost instantaneous (especially at the seizure onset) and thus can enable low detection latency. As mentioned above, EEG recorded with high spatial resolution can even enable seizure localization [74]–[78].

However, as seizures are a holistic phenomenon that cause changes in the whole body, seizure detection is possible via other biosignals as well. More specifically, epileptic seizures are caused due to dysregulations of the Sympathetic and Parasympathetic Nervous System (SNS/PNS), so heart activity which is also modulated by the Autonomic Nervous System (ANS), may provide an indicator of the ictal onset. Standardized HRV parameters can be used as markers for estimating various aspects of heart activity and used for anticipation of seizures [79]. The change in heart rate is visible in photoplethysmography (PPG) signals as well. More specifically, PPG makes use of reflected light to measure changes in light absorption caused by changes in the blood volume due to heartbeats. Thus, various algorithms and devices with PPG sensors have been tested for epilepsy detection [80]–[82].

Further, Electrodermal Activity (EDA), or also called galvanic skin response, is an autonomic marker for sympathetic skin activity, which also exhibits unique properties during seizures. In [83], authors provided a systematic review of EDA response during seizures. For example, authors in [84] recorded the surge in the amplitude of electrodermal activity after general-

ized TCS seizures. Several studies tested different ML approaches using EDA as one of the signals [85]–[87]. Even if EDA itself is not informative enough to detect seizures with high accuracy, it contains information that can help in multimodal settings. Thus, many wrist-worn devices (such as E4, Embrace2, Epilert) have an EDA sensor, among others.

Finally, electromyography (EMG) is a self-evident modality for detecting seizures with motor components. EMG measures the electrical activity of muscles, thus identifying ictal muscle contraction. It has been tested for convulsive seizure detection [88]–[90].

Signal fusion is a topic dealing with how to improve performance by using multiple biosignals, but at the same time minimize their number to lower the computational complexity and time. Several works [91], [92] tackled this topic, but there are still a lot of open questions to answer.

Features

The nature of EEG signals is very complex, non-stationary, and time-dependent [93], and thus, extracting various sets of features can significantly help in detecting patterns between seizure and non-seizure periods.

Researchers often use simple statistical features such as amplitude, kurtosis, line length, entropy, skewness, max and min values, standard deviation, energy, etc. It was shown that three simple feature sets, namely, mean amplitude, 'line length', and 'relative power', seem to be already good performers for seizure detection [94]. A lot of attention has been put on the line length feature as it consistently demonstrated good results in various studies [95]–[97]. It is a simultaneous measure of the frequency and amplitude of the signal. During an epileptic seizure, there is a sudden change in the frequency of EEG signals, which is measurable by applying frequency-domain methods, e.g., using Fourier transform (FT), or time-frequency domain methods such as discrete wavelet transform (DWT) [98]–[102]. Other studies focused on investigating non-linear features, such as entropy and its sub-types, such as Approximate Entropy (AE) and Sample Entropy (SE). For example, [103] used eight different kinds of entropy features; approximate, sample, spectral, fuzzy, permutation, Shannon, conditional, and correction conditional on raw EEG data.

Several signal processing techniques have found popularity in seizure detection, and prediction applications; Singular Value Decomposition (SVD) [104], Empirical Mode Decomposition (EMD) [102], [105], [106], Non-Negative Matrix Factorization (NNMF) [107], Principal Component Analysis (PCA) [75], and Independent Component Analysis (ICA) [108]–[110]. Many features can be extracted after the above-mentioned signal transformation techniques. Furthermore, in EEG data analysis, ICA is most commonly used to remove artifacts, while Linear Discriminant Analysis (LDA) is used to reduce the dimensions of feature sets by finding linear combinations of feature vectors.

Extracting many different features can be advantageous, but feature selection is also often a necessary part of the ML workflow. Irrelevant features make detection harder and increase complexity and time for processing. For example, [103] proposed a novel method for epilepsy

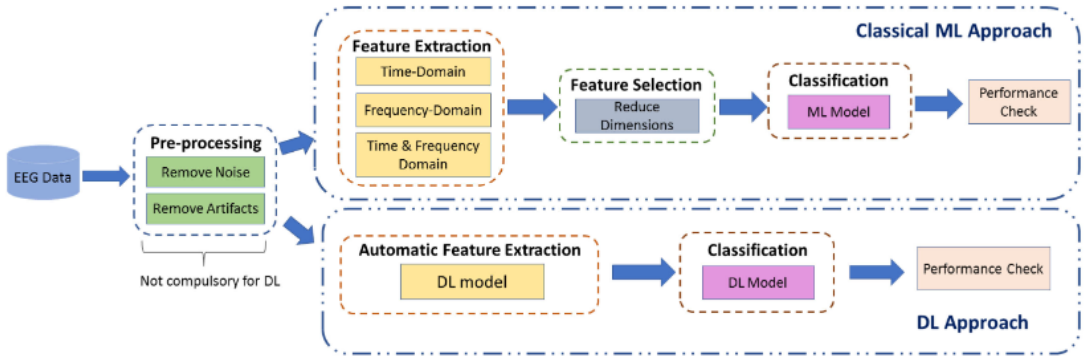

Figure 1.6: Typical machine (ML) or deep learning (DL) workflow for epilepsy detection. Illustration is taken from [29].

seizure detection by implementing a feature selection method with a random forest. In a recent paper [111], the authors propose a self-aware system where in case of good confidence, a simpler model with fewer features is used, and, only if needed, a more complex model with a larger set of features is turned on. This approach can significantly save energy and prolong battery life.

Machine learning algorithms

Four decades ago, Jean Gotman [112], [113] analyzed and proposed approaches for the usage of EEG signals for automatic seizure detection by applying different computational and statistical techniques.

Since then, a wide number of seizure detection approaches have been developed by using multiple machine learning classifiers as well as features. The most commonly used ML classifiers are SVMs [101], [107], [114]–[118], K-Nearest Neighbors (KNNs) [100], [101], [118]–[120], Decision Trees (DTs) [121], [122], Random Forests (RFs) [123]–[126] and Neural Networks (NNs) [102], [116], [127]. But researchers tested a wide variety of other approaches too, such as linear discriminant analysis [128], genetic algorithms [129], gaussian mixture models [119], systematic forest [130], fuzzy classifiers [131], swarm intelligence [77], just to name a few.

Some authors even tested ensembles of several classifiers together [102], [103], [120], [128], [131]. For example, in [131], seven different classifiers were used, while in [128], authors used nine classifiers. Here the important thing to note is that even if performance is better, the complexity and time and energy consumption of such approaches might be too high for wearable applications.

Deep Learning (DL) approaches have also recently become popular algorithms for epilepsy monitoring. Most commonly used are Convolutional Neural Networks (CNNs) [132]–[135] and Recurrent Neural Networks (RNNs) [136]–[138]. The key difference between CNN and RNN architectures is that CNNs only consider the current input while RNNs considers the current as well as the previous input. DL has the advantage that data does not necessarily have to

be filtered, and there is no need for manual feature extraction. It has the ability to discover important features through sets of interconnected layers, which can be more discriminative than manually designed features [139].

A typical ML or DL workflow is shown in Fig. 1.6. Most of the steps are optional. For example, data preprocessing steps such as data filtering and noise/artifact removal may or may not be employed as necessary in different workflows. Similarly, feature extraction is needed for standard ML approaches, whereas it is not necessary for DL approaches. Some authors also perform feature selection to reduce the number of features and thus make models faster and more lightweight. Lastly comes the model itself with the classification step and performance assessment.

Each classifier has its own merits and demerits, depending on the dataset attributes and requirements. Several recent reviews provide a comprehensive overview of research works using various ML [140], [141] and DL approaches [141], [142]. Despite this, it is generally very difficult to point out which classifier is the most effective for brain datasets. One of the reasons for this, which will be addressed in depth in this thesis, is the differences in methodologies and performance measures that make a quantitative comparison of literature impossible and meaningless.

'Black-box' vs. 'non-black-box' models

Most of the machine learning algorithms such as SVM, KNN, or NNs are, by default, 'black-box' models. This means that they only make a prediction without providing logical rules or classification procedures. This is a limiting factor for extracting sensible knowledge, and especially in biomedical applications, where patients and doctors ultimately want to know why and how the model is working to be able to interpret results properly.

Decision Trees (DTs) were the first 'non-black-box' models used for epilepsy detection [122]. Further, researchers showed that ensembles of DTs, called Random Forests (RF), could actually be more effective [123], [124] and avoid the overfitting problem of DTs [143]. A single DT discovers only a single set of logic rules from an input dataset, which may fail to predict correctly in case of highly complex patterns. RFs, on the other hand, are based on a large number of logic rules and patterns that have higher power to detect relevant patterns. Thus, many recent works focused on using RFs in epilepsy detection and have reported impressive results [144]–[146]. An added benefit of RFs is that due to their explainability, they can also be used to reduce the number of features [103] or channels [147].

Unsupervised approaches

Labeling of the seizures from EEG data is performed by experienced neurologists who are able to identify seizure patterns by visually observing EEG signals. However, this procedure is extremely costly, time-consuming, and is prone to missed seizures due to the large number of signals that have to be scanned, as well as due to complex temporal and spatial patterns. Therefore, automated techniques for labeling seizures or even just highlighting regions with a high probability of a seizure would be profoundly beneficial in diagnostics. For this purpose,

unsupervised epilepsy detection methods are needed. For example, the authors of [148] used a generative adversarial network for such a task,while the authors of [149] used a 2D deep convolutional autoencoder. Along a different line of research a recent work [150] focused on using one epilepsy dataset to highlight regions with a high probability of seizures on another dataset.

Challenges

Although many studies report impressive levels of accuracy via ML methods, the widespread adoption of commercial technology has yet to happen. The reasons for this are many and include the specificities of epilepsy itself. For example, to properly characterize epileptic seizures, recordings must be continuous, often lasting for days and leading to extremely unbalanced datasets. This imbalance must be accounted for when preparing the dataset, splitting it for training and testing, training epilepsy detection models, and reporting final performance values. Another challenge is the fact that epilepsy is a holistic phenomenon affecting many signal modalities, and thus, to get a full picture, multi-modal data is needed from several different categories of sensors. However, it is still unclear which combination of detection technologies yields the best results. How to efficiently process all this data and fuse information and predictions remains an open research topic [151]. In the end, optimal approaches may ultimately need to be individualized. Similarly, seizure activities show highly personalized patterns, which require new methods of taking general models that were developed from many subjects and personalizing them via individual patients' characteristics [152], [153].

1.2.5 State of the art in devices

In recent years, with the development of low-power, small-size wearable devices, a range of possible new applications have opened in the medical field in general, as well as for epilepsy patients [36], [154], [155]. Patients with epilepsy have expressed a strong interest in the use of new technologies in their daily lives [156].

Many of them are based on the idea of continuous non-invasive monitoring with the goal of detecting in real-time, or even predicting upcoming seizures to enable prompt reaction and prevent possible accidents. Such devices have many requirements to satisfy in order to be accepted by patients and doctors: they must be precise and reliable, but also lightweight, small, and unobstructing, with significant computational power and sufficient battery lifetime. This means that many state-of-the-art algorithms [157], [158], are infeasible due to excessive memory and/or power requirements. Another challenge is to achieve a high enough sensitivity with few or no false positives while considering the vast imbalance in data distribution (i.e., the amount of seizure vs. non-seizure data).

There is still a lack of standardization for data collection, data annotation, and testing of the algorithms and devices. In 2020, a consortium of experts from clinical research, engineering, computer science, and data analytics gathered and formed the Wearables for Epilepsy And Research (WEAR) International Study Group. They identified seven major essential components of the experimental design and reported them in [159]. Fig. 1.7, as presented in the same work, illustrates the technical environment setup within the epilepsy monitoring unit for in-hospital studies on wearable devices for seizure detection. As mentioned, video EEG is used as a standard seizure monitoring modality, which has to be recorded synchronously with the tested device. Due to the relatively small memory available on devices, devices usually need to send data wirelessly to the computer station or cloud. From there, data can be pulled, inspected, and compared with baseline EEG data. Over the whole time, a clinician needs to be observing the patient.

EEG based devices

The first devices proposed for epilepsy monitoring were based on EEG recordings using EEG caps. EEG caps are usually based on a 10-20 placement system which has very good coverage of the head but is at the same time complex to assemble, cannot be done by patients themselves, and cannot actually be used outside clinical environments. Thus, researchers tested using commercial EEG headsets. For example, EPOC+ (Emotiv) is a wireless 14-electrode headset that transmits data to a tablet, which then analyses data. In [160] authors reported a sensitivity of 39% with a precision of 71% and an F1-score of 0.51, with varying sensitivities for generalized and focal seizures (82% and 31%, respectively).

There have been efforts to lower the number of EEG channels and create smaller, more lightweight, and less visible wearable EEG devices. For example, e-Glasses [36] have been recently proposed as a small, unstigmatizing EEG device that has two bipolar electrodes built into the glasses frame. The authors tested the device using the CHB-MIT public epilepsy

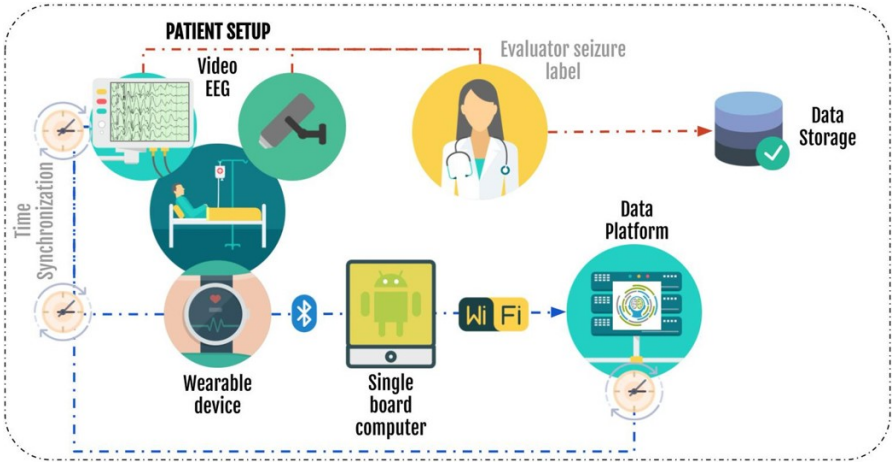

Figure 1.7: Setup for testing wearable devices for seizure detection. The picture is taken from [159])

database and have shown an impressive level of performance while providing almost three days of battery life.

Another such device is the Byteflies's Sensor-Dot (www.byteflies.com) [35], which measures two channels of EEG behind the ear, electrocardiogram (ECG), accelerometry (ACM), and EMG of the left deltoid muscle. The recorded data is transmitted securely to the cloud, with a dashboard for the clinician to interact with the data. It is a CE-certified medical device and can be adjusted to accommodate various seizure types. Researchers reported seizure detection sensitivity of 98% and False Alarm Rate (FAR), or false positive rate, of 0.91 per hour [161].

An EEG device hidden in a headband was proposed by authors in [162]. It is practical for night usage and epilepsy monitoring, as it can be quite comfortable and unobstructing. The system was validated using the CHB-MIT database and resulted in a detection rate of 92.68% and a FAR of 0.527/h.

Recently, researchers started to test the feasibility of recording epileptic EEG from behind the ear. In [163], behind-the-ear EEG channels for automated seizure detection using patient-dependent models were tested and obtained a sensitivity of 69.1% with a FAR of 0.49/day. Thus, in [37], [164] behind-the-ear devices have been tested. They have shown that behind-the-ear EEG acquires meaningful epileptic discharges similar to on-scalp EEG, and that seizure detection based on a Support Vector Machine (SVM) algorithm showed comparable seizure detection performance to whole scalp EEG.

Even more recently, the idea of in-ear devices appeared [165]–[167]. [165] is the first study that compared ictal and interictal patterns recorded with in-ear EEG and simultaneous scalp-EEG in an epilepsy monitoring unit. The authors used correlation and time-frequency analysis to quantify the similarity between ear and scalp electrodes and have concluded that in-ear EEG can reliably detect electroencephalographic patterns associated with focal temporal lobe seizures. A recent paper [38] presented an in-ear EEG system based on dry electrodes. The

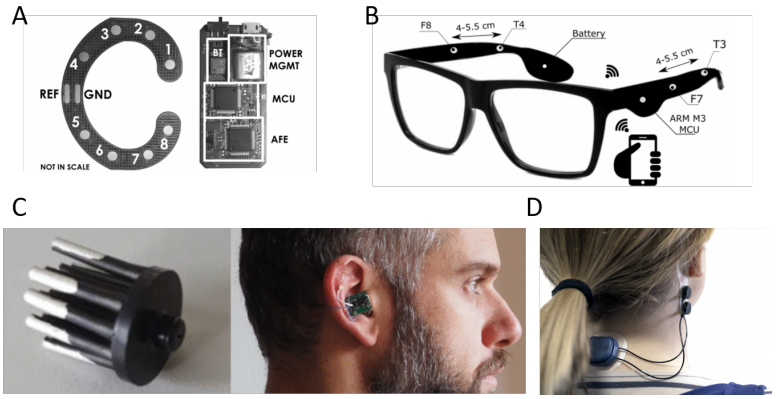

Figure 1.8: EEG based devices for epilepsy monitoring: A. behind-the-ear device proposed in [37], B. e-Glasses proposed in [36], C. in-ear device proposed in [38], and D. CE-certified device from Byteflies [161]

system can stream raw EEG data or perform data processing directly on the device. It was tested by analyzing its capability to detect brain response to external auditory stimuli but is yet to be tested for epilepsy monitoring.

EMG based devices

Surface electromyography (EMG) measures electrical activity on the surface of the skin, thus characterizing muscle contractions. EMG-based devices are sensitive to seizures with motor manifestations such as Tonic-Clonic Seizures (TCS).

For example, EMG-based seizure devices attached to, e.g. the arms, seem to be able to detect GTCS with sensitivities of more than 90% [90], [168]. EDII (IctalCare) is a wearable EMG sensor that is worn on the upper arm with a sensitivity of 94% in the detection of TCS [90]. Most false positives were reported due to muscle activation during exercise. The SPEAC monitor (The Brain Sentinel) is very similar to the EDII, and showed a sensitivity of 95% with a very promising F1-score of 0.95 and a low FAR of 0.017 in the detection of TCS [168].

Accelerometry based devices

EpiCare Free (Danish Care Technology) is a wrist-worn ACM-based seizure detection device. It was tested in a multi-center clinical study and showed a sensitivity of 90% with a FAR of 0.2/day for detecting bilateral TCS. In [169], the authors assessed that users' acceptance of the device would be quite high.

SmartWatch (SmartMonitor) [170] is a wristwatch that continuously monitors movements using an accelerometer and instantly alerts connected family members and caregivers upon the onset of repetitive, irregular shaking motion. Thus, it can alert caregivers when the patient is having a TCS, and the patient can also push an alert button during an aware focal seizure. Authors reported a sensitivity of 16%. As accelerometers are integrated into a wide range of devices, researchers also tested devices such as Nintendo's Wii Remote [171] or an iPod Touch [172].

Figure 1.9: Accelerometry and/or EMG based wrist devices for epilepsy monitoring: A. EpiCare Free (Danish Care Technology) , B. Evarion Biovotion (BioFourmis), C. EDDI (IctalCare), D. SPEAC (The Brain Sentinel)

It is of course important to note that ACM-based devices, similar to EMG-based ones, can detect only the subset of seizures that include motor components.

Multimodal devices

Studies have also shown that there are significant changes in the heart rate (i.e., tachycardia or heart rate variability) [173], [174] that may represent an additional information source for improving detection and prediction. Heart rate can be measured using ECG or estimated from PPG, which is easier to measure as sensors can be on the fingers, wrist, or even ears. Hearth rate can be used to detect seizures, but here a personalized detection algorithm is necessary, as they perform significantly better than patient-independent models [153].

The Epilert bracelet (Epilert, www.epilert.io) is a waterproof smart wrist bracelet connected to a mobile application via Bluetooth. It tracks several biosignals such as heart rate, skin temperature, EDA, and movement. Furthermore, it also has built-in machine learning algorithms to detect seizures, and in case of a seizure, it alerts the caregiver and sends the patient's location and condition. It was reported to have a sensitivity of 91% and an F1-score of 0.80 for the detection of tonic and TCS seizures [175]. A special dashboard for medical doctors is also available to visualize all data related to patients. Data like frequency and duration of seizures, biosignals data, and other statistics collected through the bracelet are available.

The Nightwatch (www.nightwatchepilepsy.com) is a multimodal device measuring ACM and PPG signals. It is worn on the upper arm to detect generalized nocturnal tonic and TCS seizures. The authors of [176] reported that while the heart rate measurements were responsible for false positives, the ACM signals were critical for detecting true positives. The device gave an accurate sensitivity of 86% and an F1-score of 0.62.

The Empatica E4 (www.empatica.com/research/e4/) wristband contains ACM, EDA, temperature, and PPG sensors. It has been validated in multiple studies, in different patient cohorts with TCS, and in both inpatient and outpatient settings. Overall, it has a sensitivity between

Figure 1.10: Multimodal devices for epilepsy monitoring: A. The NightWatch records ACM and PPG, B. Epilert device records HR, SKT, EDA and AMC, C. Empatica E4 records EDA , SKT, PPG and ACM, D. Empatica Embrace2 records only EDA and ACM.

92 and 100% [85]. It was primarily for research purposes, so Empatica developed a new consumer-focused Embrace2 (www.empatica.com/embrace2/) bracelet with FDA approval. It also combines EDA and ACM for the detection of TCS. While having an accurate sensitivity of 94%, it suffers low precision of 10%, resulting in a low F1-score of 0.18 [177]. The authors showed that the performance of the device was lower if only one modality was used instead of using both, suggesting that multimodal sensors provide better sensitivity and lower FAR rates.

Important characteristics of wearable devices

In the end, despite not having many devices well-established on the market, the field is evolving fast. Thus, it is important to be aware of important aspects of wearable devices that have to be taken into account for epilepsy monitoring. Here we list the most important factors:

- **Medical certification:** Certification as a medical device is an important factor that significantly simplifies ethical approval protocols for using the device in studies. However, certification is a long, complex, and expensive procedure. Thus, today not all devices have medical certification. Based on our knowledge, *Empatica E4* and *Empatica Embrace* have FDA approval, while *Sensor Dot*, *EDII*, *Epi-care* and *Nightwatch* are CE medically certified.
- **Signals recorded:** Recording several different biosignals simultaneously is extremely valuable and increases the utility of the data recorded and the range of studied research questions. But there are also challenges for recording multiple biosignals at the same time, such as physical practicalities, different sampling rates, synchronization questions...
- **Operating mode:** Devices can be used not only to record data and store it locally on internal memory, but they can also be used to stream data online over wired or wireless channels or even upload to the cloud. Wireless streaming or uploading to the cloud has many advantages, but also requires a significant amount of energy and thus reduces battery lifetime. In the end, new low-power machine learning approaches are bringing

on-device analysis and detection closer to reality.

- **Battery lifetime:** Battery lifetime also plays an important role in the utility of a device. With current battery technology, the battery life of smaller devices or those that employ online data streaming is usually measured in hours, whereas devices with offline, on-device data storage can sometimes be active for days without the need to recharge.
- **Device form factor:** Wearable devices face the challenge of being as small, lightweight, and invisible and nonstigmatizing as possible. From studies on patients' experience with wearables, the stigma that wearing a device poses plays a big role in whether patients will use the device or not [178]. Other important factors are also the form factor, comfort, and stability on the body. Furthermore, some devices are designed for specific locations, whereas others are more flexible in terms of placement on the body.
- **Purpose of device:** Devices can be designed as a consumer-grade product for end-users. In this case, they are rarely open for access to raw data, modifications, or alternative applications. On the other hand, research devices allow more flexibility and raw data access and enable a broader range of applications, which is very important for clinical and research studies. Research devices are, however, often more expensive and can be more cumbersome and uncomfortable to wear.

To conclude, several Wearable Seizure Detection Devices (WSDDs) are available on the market (Empatica Embrace2, Epilert, Epi-Care, Nightwatch, SPEAC, and EDII). The limitation is that they are commercially available but not yet in use as a practice. Moreover, as they are mainly based on ACM and EMG, they are mostly developed and validated on seizures with strong motor components such as TCS seizures, which represent only a portion of epilepsy seizures, and usually fail to detect other types of seizures [155], [168].

As TCS seizures tend to follow a stereotypical pattern, the automated seizure detection algorithms are patient-independent, and there is no need for individual seizure personalization [32]. However, a major challenge facing the detection of focal seizure manifestations is the large inter-individual variations, which will require a personalized, patient-dependent automated seizure detection algorithm [32]. Patient-specific algorithms are needed to increase sensitivity and decrease false-alarm rate [99], [163], and thus, WSDDs to detect seizures other than TCS are currently in development and not available on the market.

Generally, in the future, business models and the landscape of using WSDDs has to be developed further to be applicable in generalized scenarios. Automated seizure detection algorithms could reduce the vast amount of collected data, by creating a selection of data with high probability of containing seizures. In this way the neurologists will be able to assess the recorded raw data by rapidly scrolling through the detected potential seizures, retaining true positives and discarding false positives [32]. Moreover, the integration of recorded data and algorithmic detection in the electronic health record will be of importance. This will only be possible if neurologists, engineers, and medical device companies find ways to improve collaboration, enabling the free flow of data and advanced algorithms and new approaches to seizure detection and prediction [179].

1.3 Hyperdimensional computing

1.3.1 Basics of hyperdimensional computing

Hyperdimensional (HD) computing is a promising new ML approach inspired by neuroscience [180], which posits that large but simple networks are fundamental to the brain's computational power. More precisely, neuroscience results suggest that the brain's computation is based on high-dimensional randomized representations of data rather than scalar numerical values [180]. Similarly, HD computing is based on representing data as vectors with very high dimensionality (usually > 10,000 values [180]).

In the 80's philosophers and cognitive scientists questioned the concept of connectionism. Connectionism is an approach in the field of cognitive science that hopes to explain mental phenomena using artificial neural networks. It also uses a wide range of algorithms and techniques to build more intelligent machines. It is based on the theory that states that cognition can be modeled by simultaneously occurring, distributed signal activity via connections that can be represented numerically, where learning occurs by modifying connection strengths based on experience. Neural networks are by far the most commonly used connectionist model today.

Smolensky [181] formalized the idea that a set of value/variable pairs can be represented by binding using tensors products, i.e. by computing the product of a feature of a variable and a feature of its value. It served as the basis for the association operation. It enabled the ability to 'bind' together sets of information. The problem with the tensor product is a combinatorial explosion problem, where binding two vectors of N elements results in a vector with N^2 elements, and so on. There have been various proposal on how to stabilize the dimension of the vectors no matter how many information sets are associated together.

Today, systems that are based on these representations come under different names such as Holographic Reduced Representation (HRR), Hyperdimensional Computing (HDC), Vector Symbolic Architectures (VSA) [182], Binary Spatter Code (BSD) [183], Binary Sparse Distributed Code (BSDC) [184] or Multiply-Add-Permute (MAP). HRR use circular convolution, BSC use Haddamard product, MAP used permutation to limit the explosion of the vector dimensions. Solving the problem of combinatorial explosion made it possible to use vectors of any dimension. Despite their differences, all these approaches have in common the use of high dimensional vectors, distributed information representation, vector orthogonality, and learning through superposition.

Algebraic properties
Calculating and learning with such large vectors is based on a few specific algebraic properties that are crucial for learning and inference.

- For vectors of dimension N, the space of such vectors contains on the order of 2^N nearly-orthogonal vectors.

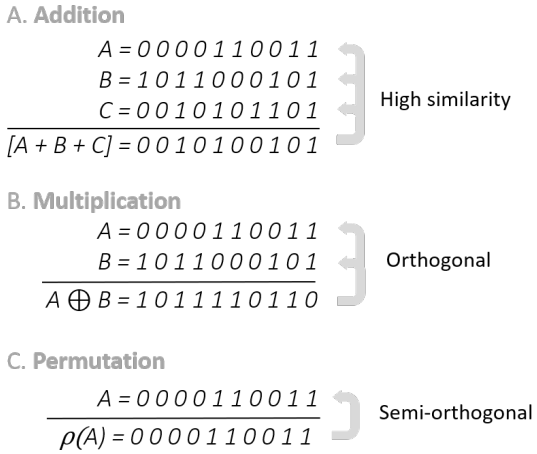

Figure 1.11: Basic operations with HD vectors: A. Addition, B. Multiplication and C. Permutation. The similarity of the resulting vector with initial vectors is also illustrated.

- Vectors can be degraded by up to 30% and still be closer to their original form than to any of the other vectors in the space [180].
- Thus, any randomly chosen pair of vectors (with dimensionality in thousands) are with high probability nearly orthogonal.
- When summing two or more vectors, the result will be with high probability more similar to the added vectors than to any other randomly chosen vector.
- The product of two vectors results in vector that is orthogonal to the multiplied vectors.
- Vector multiplication and addition are associative and commutative, and multiplication distributes over addition.

Basic hypervector operations

Unlike many traditional neural networks, HD computing does not rely on backpropagation or other compute-intensive learning algorithms. The HD approach is based on several simple operations:

- **Bundling** is the element-wise addition of all vectors. For binary vectors this is done by bitwise addition. To regain binary vectors, normalization using majority voting is usually performed after addition. The resulting vector is similar to added vectors. Bundling is illustrated in Fig. 1.11-A.
- **Binding** is element-wise multiplication, and is used to associate two hypervectors. For example, vectors A and B may be bound together to form $X = A \oplus B$, which is approximately orthogonal to both A and B. For binary vectors, the bitwise XOR operation is used as $\oplus$. This is illustrated in Fig. 1.11-B.
- **Release** is the inverse of multiplication. It is used to extract information from a 'bound' vector. For example if we have $X = A \oplus B$, to release B from X, we need to bind it with A: $A \oplus X = A \oplus (A \oplus B) = B$ as $A \oplus A$ cancels out.
- **Permutation** shuffles a vector by a given number of positions. The resulting permuted

vector is quasi-orthogonal to the initial vectors i.e., the normalized Hamming distance in the case of binary vectors is close to 0.5. Often, a circular shift is used as a permutation for hardware friendly implementation.

More specifically, the basis of learning in classification tasks is in vector bundling. Vectors belonging to the same class are summed together to form a 'model' vector that will represent each class. For example, to represent a class consisting of three samples A, B and C, a vector to represent the class is simply learned as $S = [A + B + C]$. If each sample is represented by several features, e.g. $F1$, $F2$ and $F3$, and respective feature values, e.g. $V1$, $V2$ and $V3$, the HD vector representing these samples may be encoded as $D = [F1 \oplus V1 + F2 \oplus V2 + F3 \oplus V3]$.

Similarity between vectors
Inference is based on a similarity metric between hypervectors representing data samples and all classes' 'model' vectors. The most commonly used similarity metrics are:

- **Hamming distance**: This metric is applicable only for binary vectors whose elements are usually 0 or 1. It measures the percentage of bits with different values. Thus, Hamming distance has a value of 0 when vectors are identical, 0.5 when vectors are orthogonal and 1 when vectors are diametrically opposite.
- **Cosine distance**: This measure is applicable for binary as well as non-binary vectors. It measures the angle between vectors, and thus the cosine distance is 1 between very similar vectors, and 0 when vectors are orthogonal.

Other distance metrics such as Euclidean distance, dot product (e.g., in MAP) or overlap (e.g., in BSDC) can be used too.

An important property of HD computing that makes it highly interesting for wearable applications is its potential for parallelization. Namely, element-wise multiplication and addition are easily parallelizable operations that can be performed efficiently (in principle, in constant time). Moreover, for computationally lightweight applications, binary vectors can be used. This enables lower memory requirements and utilizes simpler operations, such as bitwise SUM and XOR.

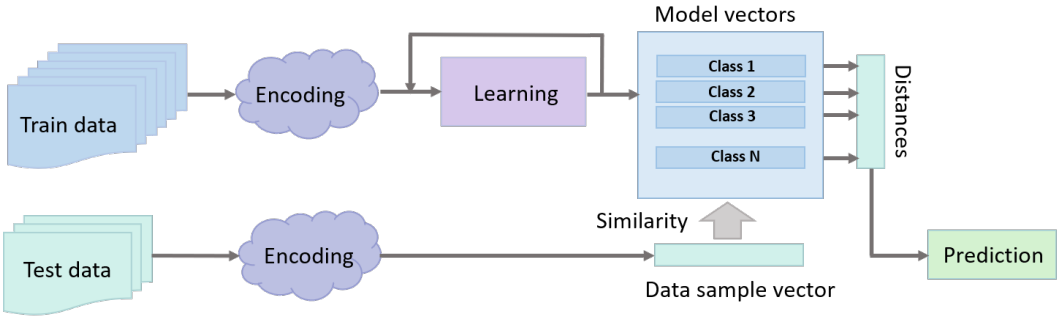

Figure 1.12: Simplified HD computing workflow.

1.3.2 HD computing workflow

The HD computing framework consists of three stages: encoding, training, and querying, as illustrated in Fig. 1.12. Encoding is the step in which HD vectors are initialized, and data/features are encoded to HD vectors. Training consists of summing up (bundling) all vectors coming from the same class. Then, during the querying step, labels are inferred based on the similarity of the current HD vector with the prototype vectors representing each class.

As mentioned, HD learning starts by encoding data and its relations to HD vectors. As illustrated in Fig. 1.13, baseline vectors representing different features and their values are combined during the encoding stage to get one vector. This vector represents that specific data sample, instead of a feature set, as in other ML approaches. For time series data, as shown in Fig. 1.13, data is discretized into windows of duration W_{len} that are moved in steps of W_{step}, i.e., for every W_{len} of data, features are calculated and encoded into an HD vector representing that discrete window of data. Researchers have tested using permutation information to encode time, i.e., signal time delay.

The training phase is relatively simple and consists of summing (bundling) all vectors from the same class to one prototype/model vector representing each class. As explained, the summation of the vectors is usually done by bit-wise summation, followed by majority voting normalization at the end. More specifically, if a majority of the vectors on that bit position have a value 1, the final value is 1, otherwise 0.

In the end, for inference, a vector representing the current data sample is compared with prototype vectors of all classes, and the label of the most similar one is given as output. The most common measure of similarity is the Hamming distance in the case of binary HD vectors (elements are 0 or 1), but cosine or dot products can also be used in the case of integer or float HD vectors.

Traditionally, HD computing classifiers have been based on single-pass learning and with a single-centroid model vector per class, meaning the training data is passed through only once, and only one model vector is generated per class in the training set.

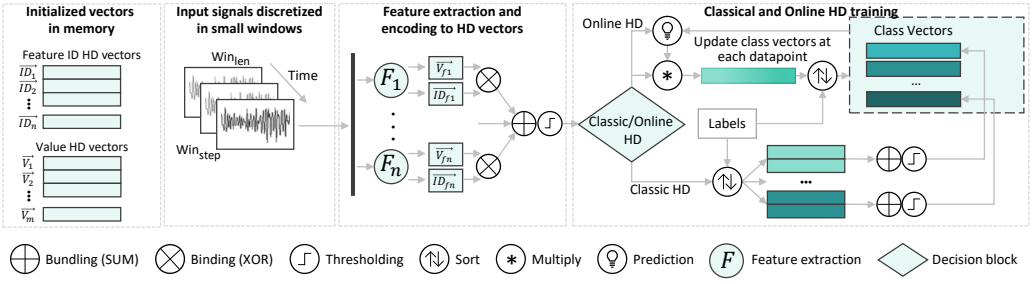

Figure 1.13: The HD workflow for training classical and online HD models. Online training differs in that the class vectors are updated after every datapoint by multiplying its similarity to the target class by the vector before accumulating it into the class.

Data encoding

The first step when performing training or inference via HD computing is to encode raw input data into the HD space as hypervectors. Lets say that we have set of N features that can have various values. To encode features with their values to one vector there are usually two steps: 1) initializing hyperdimensional basis vectors, that are later used for 2) the encoding procedure, that combines initialized vectors using the presented arithmetic operations to get a vector representing one data sample. In this case for each of N features, one vector per feature has to be initialized. For feature values, values have to be normalized and discretized to a fixed number of bins. This then enables initialization of basis vectors for each discretized value.

The above described vector initialization can be performed in various ways. For example, if there is no specific relationship between the values represented by these vectors (e.g., in the case of the N different feature vectors) each vector can be randomly initialized. If there is relationship between values (e.g., in the case of the discretized values for each feature), the relationship between values can be also mapped to similarities between vectors. This can be done by, for example, flipping a predefined number of bits of the vector to get the next (neighbouring value) vector. This is sometimes called encoding using 'scale', 'thermometer' or 'level' initialization, and will be more explained later.

Finally, encoding performed bundling feature ID vectors and their corresponding values is also called with different names. For example, in [185] authors call it 'record-based' encoding, while in [186] it is referred to as ID-Level encoding. In any case, it is based on a simple idea that each feature F has a representative 'ID' vector, while all values F can take on are discretized into a fixed number of bins, with each bin having its own representative 'Value' vector. ID vectors ($\vec{ID}$) are randomly generated, while Value vectors ($\vec{V}$) may be randomly generated or, as in [187], generated using a linear scaling method so that vectors representing similar values are also similar. Using the predefined $\vec{ID}$ and $\vec{V}$ vectors, each data point is encoded to a hypervector $\vec{H}_i$ by binding each feature vector $\vec{ID}_{fi}$ with the bin value vector $\vec{V}_{fi}$ corresponding to the value of the feature. Encoded features are then accumulated as formulated in Equation 1.1. The final summed values are normalized using majority voting to

regenerate the binary vector.

$$\vec{H}_i = \left\lfloor \sum_{f_i} \vec{ID}_{fi} \oplus \vec{V}_{fi} \right\rceil \tag{1.1}$$

'N-gram-based' encoding is another option for encoding with fewer initialized vectors [185]. In this approach, only values of features ('Value' vectors) are initialized as vectors, while encoding of the feature ID to which they are related is done via the permutation operation as formulated in Equation 1.2. For example for the first feature the 'Value' vector of that feature is not rotated, for the second feature the 'Value' vector is rotated for one position, for third feature for two positions and so on.

$$\vec{H}_i = \left\lfloor \sum_{f_i} \vec{\rho}^{i-1}(\vec{V}_{fi}) \right\rceil \tag{1.2}$$

This encoding requires less initialized vectors, and might be simpler to implement, but has been reported to show lower performance [188].

Encoding is one of the most essential parts of learning in HD computing, as different types of data or use cases might have different optimal encoding strategies [189], [190]. Thus, designing new approaches to effectively and efficiently represent data in the form of vectors is a relevant research topic. Many papers discuss issues with encoding and propose new approaches to it. Part of this thesis will also discuss encoding approaches for spatio-temporal data such as EEG or EMG.

Training

Once encoded, the data can be processed in different ways. The most basic approach is the so-called classical HD training method, which utilizes the single-pass accumulation of encoded vectors belonging to the same class into a class vector.

On the other hand, recent literature has proposed various improvements to classical HD training, such as iterative learning [191], and progressive, online ('OnlineHD') learning [192]. Iterative learning is based on passing through the data several times. Online training, on the other hand, improves accuracy by, instead of treating all data points as equally important, multiplying new data points by their similarity to current class vectors before being accumulated, as illustrated in Equations 1.3 and 1.4:

$$\vec{M}'_C \leftarrow \vec{M}_C + (\delta_C)\vec{H} \tag{1.3}$$

$$\vec{M}'_W \leftarrow \vec{M}_W - \gamma(1-\delta_W)\vec{H} \tag{1.4}$$

where $\vec{M}_X$ is either the correctly or incorrectly classified class vector, δ_X is the distance from

the class vector (a lower distance means more similarity), and γ is the learning rate. In this way, highly common class patterns are not allowed to saturate the class vectors, thus improving sensitivity to less common patterns. Online HD training is also applicable in continuously learning wearable devices, as it is capable of integrating new data in real time. The workflow is described visually in Figure 1.13.

1.3.3 Advantages of HD computing

Explained change in the data representation, from individual scalars to distributed hyperdimensional vectors, brings various advantages from a learning and hardware implementation perspective. From a learning perspective, it opens new paths for:

- **Continuous online learning:** HD computing has the advantage being able to update model vectors with new data samples as they are obtained, instead of having to retrain the model from scratch. This makes continuous learning on devices possible [193], [194]. It also means that there is no need to keep all previous data in the memory to be able to retrain the model, which significantly reduces memory consumption.
- **Distributed learning:** As model vectors are calculated via the summation of individual model vectors, training of the global model can be distributed on several actors (GPUs, servers or individual devices) which can then send their models to update the global model [195].
- **Semi-supervised learning:** The authors of [191] describe approaches to label unlabeled data with high confidence and thus iteratively expand the training set.
- **Multi-centroid learning:** Instead of assuming that each class has only one model vector, in this thesis we propose a clustering approach to create several 'centroids' per class [196].

From the hardware implementation perspective, HD computing has many advantages as well:

- **Robustness to noise:** Due to their distributed data representation, HD vectors are more robust to noise, especially in a low power devices [192], [197]–[199].
- **Highly parallelizable:** Elementwise multiplication and addition are easily parallelizable operations that can be performed efficiently, sparking interest for FPGA implementations [200], designing efficient accelerators [201], and in-memory computation [202].
- **High energy efficiency:** Researchers have showed HD computing's lower energy requirements [203]–[205].
- **Potential for IoT and wearable devices:** All previous aspects make HD computing an interesting choice for implementations on small, low power, and memory and computationally restricted devices.

Finally, when talking about wearable biomedical applications, it is important to note a few more interesting properties of HD computing systems:

- **Reduced data preprocessing requirements:** Vectors have expressive power to map even

raw data or data trends to vectors [206], thus lowering the need to high quality data preprocessing, and filtering.

- **No need for expert knowledge:** Similarly as above, some research has shown that it is possible to avoid highly specific domain features that require expert knowledge, and encode raw data trends and achieve good performance [190].
- **Potential for one-shot learning:** Ability to learn from only a few samples of data is extremely valuable for biomedical applications. HD computing has been shown to have potential even in few/one shot learning scenarios [206].
- **Possibility for knowledge transfer between models:** Models from different individuals can be easily combined or compared. This also allows creating generalized models whose knowledge can be passed on to individualized models. This will be in more details explored in this thesis.
- **Interpretability:** HD computing can enable backtracking contributions of individual features and channels to the decision, thus opening a potential for the interpretation of results, which will also be discussed in this thesis.

1.3.4 State of the art of HD computing

Applications of HD computing

Even though HD computing was first proposed in the 1990s, only Kanerva's 2009 overview [180] improved accessibility and interest by illustrating its advantages and potential applications for neuroscience, computer science, mathematics, and engineering. Since then, HD computing has been applied to a broad range of domains and uses a broad range of input data such as letters, time series, images, or even graphs and trees.

Letters

HD computing is of high interest for natural language processing (NLP) applications, where it can be used for language recognition, topic extraction, sentiment analysis, and more. For example, HD computing has been tested for language recognition, where the idea was to model languages as three-letter sequences from which a language can be detected [207], [208]. The key idea is that the n-gram statistic is replaced by a distributed representation in the form of vectors. A similar approach was also used for clustering car trajectories that are essentially sequences of locations. The authors of [209] proposed an unsupervised incremental learning approach for road traffic congestion detection and dynamic profiling over time. Other researchers used DNA base sequences for species classificaiton [210], achieving better than state-of-the-art accuracy. In [198], the authors designed an accelerator architecture which effectively parallelizes the HD-based DNA pattern matching while significantly reducing the number of memory accesses. Finally, the authors of [211] demonstrate the usefulness of HD computing-based embedding for NLP, which is also highly time and memory efficient.

Signals

Speech recognition is one of the applications with high interest for fast and energy-efficient processing to be implementable on small battery-powered devices. Thus, in [187], authors designed an HDC-based system to classify 26 letters from a spoken dataset. They even test a combination of HDC learning with a small 2-layer NN and show that HD encoding and training as an input for an NN achieves better performance, speedup, and higher energy efficiency than an NN-only system. HD computing has also been tested for industrial applications, e.g., for fault isolation in a power plants [212], or signal modulation for robust communications on low-power devices [213]. The biggest amount of signal encoding to HD vectors has been in the biomedical domain. These signals are usually time-series signals, which will be discussed, in detail, in the next section 1.3.5.

Images

HD classification has been used for letter/character recognition [214], [215], demonstrating good performance. The authors of [216] proposed an encoder based on the permutation coding technique that allows taking into account not only detected features but also the position of each feature in the image, making the recognition process invariant to small displacements. They tested it on character recognition as well as facial and object recognition tasks. Several papers applied HD computing for facial recognition [192], [217]. Further, as image classification is an important topic of autonomous driving, the authors of [218] encode a camera event-based stream as HD vectors. They demonstrated the ability to have dynamic world perception, which creates an opportunity for real-time navigation and obstacle avoidance.

Graphs and trees

Methods for encoding entire knowledge graphs have been proposed in [219], and has recently gained more popularity in [220], [221] and [222], in which authors proposed encoding complex graph structures while supporting both weighted and unweighted graphs. Similar ideas have been used for encoding hierarchical tree-like structures [223].

Improving the HD workflow

In order to improve the learning capabilities as well as memory and computation efficiency, research is ongoing to implement improved workflows for HD computing. For example, in [224], researchers tested performance drop when replacing floating point hypervectors with binary or tertiary vectors. To compensate for the accuracy degradation caused by quantization, the authors proposed to iteratively retrain models to restore accuracy. Further, due to high dimensionality, computing with hypervectors can be quite memory intensive for embedded devices with limited resources. Thus, in [225], the authors investigated ways to reduce the high dimensionality of hypervectors without sacrificing accuracy. The proposed approach, called 'CompHD', reduces the model size by 69.7% while achieving 74% energy improvement. Similarly, the authors of [226] propose a kernel-base binary encoder approach to build HDC models that use binary hypervectors of dimensions that are orders of magnitude smaller than those found in the state-of-the-art HDC models, yet yield equivalent or even improved

accuracy and efficiency.

Further, as mentioned, single-pass learning does not always lead to satisfying accuracy. Thus, iterative training may be logical potential solution. However, a lack of controllability of training iterations in HD classification may also result in slow training or even divergence. To solve this training issue [227] proposed a retraining approach called 'AdaptHD'. In later versions, this idea is also called 'OnlineHD' [192]. As explained in 1.3.2, while passing through data iteratively, instead of treating each data points as equally important, new data vectors are multiplied by the weight defined by their distance to current class vectors before being accumulated, as illustrated in Equations 1.3 and 1.4. In 'AdaptHD', authors even adaptively changed the learning rate γ, testing different approaches to changing it. For example, the obvious option was to have a higher value at the beginning to get faster close to optimal performance and then lower γ for better fine-tuning.

The authors of [191] have designed 'SemiHD', an approach for performing semi-supervised learning from a dataset with a small portion of labeled data and a large portion of unlabeled data. More specifically, they test the approach to label unlabeled data by using HD prediction and subsequently measuring the classification confidence for each sample. In the case of high confidence, a small percentage (defined by a so-called 'expansion rate') of unlabeled data is given a label. In an iterative manner, the amount of labeled data is increased, and the process is stopped once the accuracy saturates. The authors demonstrated that this approach can, on average improve the classification of supervised HD classification by 10.2% on the tested datasets. A recent paper [228] introduced a novel approach for unsupervised learning based on fast learning from only a few input samples and a single vector operation learning rule implementation.

Furthermore, HD computing is also highly interesting for federated learning; thus, researchers proposed in [229] a privacy-preserving distributed framework. Moreover, recently, authors started combining HD computing encoding with neural networks [230]. These areas still have vast research potential.

HD computing is still in its infancy, and thus there are still many research directions that have not been fully explored. Some of these include: optimal feature extraction and encoding, hardware-friendly similarity metrics, multi-class hypervector systems, hybrid systems that combine HD computing with traditional ML methods, and hardware acceleration, to name a few.

1.3.5 Wearable healthcare applications with HD

As mentioned previously, the main advantages of HD computing are energy efficiency, a lower need for significant data preprocessing, scalability, analysability, no need for expert knowledge for feature and model construction [190], and the potential for few/one-shot learning [206]. Consequently, different approaches have been proposed and show high potential

for biomedical applications.

More specifically, HD computing has been applied to different biomedical signals: electromyogram (EMG), electrocardiogram (ECG), electroencephalogram (EEG), respiration (RSP), galvanic-skin response (GSR), and more. Here we give examples of HD applications with different types of biosignals.

EMG
Hand gesture recognition has been a common task for which different HD computing models were tested. For example, the authors of [231] developed a few-shot learning system that achieves an average classification accuracy of 96.64% for five gestures, with only a 7% degradation when training and testing across different days. The authors of [194] presented a wearable EMG gesture recognition system based on the HDC paradigm, running on a programmable Parallel Ultra-Low Power (PULP) platform, and have achieved an 85% average accuracy on 11 gestures. Varying contraction levels of muscles is a big challenge in EMG-based gesture recognition, so the authors of [232] designed HD-based classification models that are both robust to these variations and able to recognize multiple contraction levels.

EEG
EEG data can be used for detecting a user's intention through analysis of EEG error-related potentials. For example, in [233], [234], the authors reported results better than classical ML algorithms. Furthermore, as will be shown in this thesis, EEG is the most important biosignal for epilepsy monitoring. Thus, many researchers tested HDC approaches for epileptic seizure detection, both with iEEG [206] and EEG [204] signals.

Multi-modal data
ECG and RSP signals were used to analyze cardiorespiratory synchronization during paced deep breathing [235]. HD computing has also been used to test multi-modal sensor fusion. For example, in [236], the authors tested emotion recognition from GSR, ECG, and EEG. In [237], the authors proposed a highly energy-efficient HDC processor for wearable multi-modal emotion classification. The authors of [238] built a model to classify septic shock up to three hours before the onset. They used HR, RSP, blood pressure, and blood test results as input data.

Applications of HD computing for wearable healthcare devices are becoming more broad and diverse due to the various advantages of HD computing as listed in 1.3.3.

1.3.6 HD computing for epilepsy detection

The first work that applied HD computing to epileptic seizure detection [206] relied on mapping Local Binary Patterns (LBPs) to HD vectors. LBPs map a sequence of data samples onto a small binary array, relying solely on whether the data amplitude increases or decreases. The authors encode LBPs into HD vectors and test the approach on iEEG data (SWEC-ETHZ) from 16 patients from the Inselspital Bern epilepsy surgery program. They focused on testing

one-shot learning or learning from as few seizure instances as possible. They showed that the algorithm learned from one to two seizures for most patients and achieved perfect specificity and sensitivity. Still, it was based on iEEG, which is not realistic for wearable applications. Further, the authors utilized postprocessing that was adjusted to individual subjects, which is also not ideal for wearable applications. Authors applied the same approach to also identify ictogenic brain regions [239]. More specifically, the algorithm identifies the ictogenic brain regions by measuring the relative distances between the learned seizure and non-seizure prototype for each electrode. If the difference between them is high, it means that that the electrode captures seizure activity. Identification was done at two levels of spatial resolution, the cerebral hemispheres and lobes, which can be important information for clinicians. Information can be used for planning surgeries or improving post-surgical seizure control.

Later, the authors extended this work in [240] by also using the mean amplitude and line length features, besides LBP, to describe data. Each feature formed its own prototype vector for every class and acted as a standalone classifier. Then, the predictions (more precisely, vector distances) are fed into a single-layer perceptron with three neurons to decide the final prediction. The authors showed better performance and lower latency on the same dataset than the previous paper [206]. These works also compared and showed advantages over other state-of-the-art algorithms for epilepsy detection regarding performance, memory, and computational requirements.

In another paper, HD computing was applied to EEG data rather than iEEG, which is more viable for continuous long-term monitoring [204]. The authors also compared HD computing with different standard state-of-the-art ML approaches (KNN, SVM, regression, random forests, and CNN). For KNN, SVM, regression, and random forests, they used 54 different features from [241] and [36], while for the HD computing approach, raw amplitude values were encoded into HD vectors. More precisely, they normalized data and mapped amplitudes of all samples to the corresponding HD vector. The authors used the CHBMIT database [242], [243] and reported that the HD approach surpassed the performance of all other approaches.

Both aforementioned papers present promising results for the application of HD computing to epileptic seizure detection, but results cannot be directly compared. Namely, data preparation (filtering, train/test split), segmentation (size and step of discretized windows), mapping to HD vectors (how channel and time information is encoded), as well as performance metrics (sensitivity, recall, specificity, number of false positives) are different between papers. Also, based on the methodologies used in these papers, it is still unclear how well HD computing would perform on real-time wearable data. Specifically, in [206], [239], [240] iEEG data is used, which is not realistic for a wearable device, and only a small subset of data was used. On the other hand, the authors of [204] used EEG data, but temporal relation was lost as data samples were shuffled.

Furthermore, reported performance is very sensitive to the ways it is measured. For example, some works split the data into independent discretized windows and report a performance

based on independent window labels, while others observe windows as a time sequence and also take seizure dynamics into account by performing additional post-processing steps to smooth the labels. Further, performance measures can be based on the level of seizure episodes or considering the whole seizure duration. This is an ongoing topic of discussion, addressed in recent papers [244], [245], specifically applied to epilepsy detection.

All this means that there is still a need to further assess the performance of HD computing for wearable epilepsy detection. Thus, in this thesis, we try to perform a systematic assessment of the HD computing framework for the detection of epileptic seizures. In particular, as part of this thesis, we design and compare different feature encoding strategies and different learning approaches by evaluating them in a comparable way using a common workflow (i.e., with the same preprocessing setup and identical performance measures).

1.4 Contributions

1.4.1 Challenges for HDC for wearable healthcare applications

A lot of researchers are using traditional supervised machine learning methods for epilepsy detection and prediction: random forest [246], support vector machines [247], Bayesian analysis [248], artificial neural networks [247]), and more recently deep learning algorithms [249].

But there are several challenges when applying traditional ML methods in applications where algorithms have to be able to run online, in real-time, and perform detection while subjects are living their ordinary lives. This means that algorithms must also be computationally and memory lightweight enough to be implementable on small wearable or implantable devices. This is not the case with most traditional ML algorithms, especially if there is a need to (re)train models online and not just run inference on devices. Online learning or the ability to update/retrain models in real-time as new samples are acquired is not yet a well enough researched topic in the ML domain. Namely, offline training and slow (iterative) training time prevent traditional ML approaches from performing online and incremental learning from new seizure occurrences. Hence they cannot be quickly adapted to new patients, new environments, and new seizure dynamics. Most algorithms require retraining using all historical data, which requires large memories to store all data.

Furthermore, today's IoT, cloud, or fog devices can profit significantly from communicating between themselves or with a global central server. For example, in the case of epilepsy, each device can train and run individual personal models. The central cloud could gather this information, analyze similarities between them, and update individual devices with the general knowledge that can help them in personalized detection. This interplay between general and personalized knowledge and models can be incredibly valuable, especially in the medical domain, but it is still not well-researched. It is also restricted by the ability of current ML models to transfer knowledge between models and perform federated learning.

All these challenges can be addressed by HD computing. Namely, as mentioned before, HD computing relies on a set of simple, parallelizable operations that make it energy efficient and friendly for wearable and implantable devices. The basic learning approach of HD computing is actually a form of online learning, implementable on devices, without the need to store all previous historical data to be able to retrain models. Furthermore, as it will be shown later, data encoding in the form of long distributed vectors rather than complex networks or structures makes them easily comparable. This also allows the creation of additional knowledge by comparing them and then further using and transferring this knowledge between models. Ultimately, there is a huge potential for interpreting models and their predictions.

However, despite promising initial results of applying HD computing to biomedical applications such as epilepsy, there are several challenges and topics that need to be addressed in more detail to be able to understand the potential and limitations of the HDC approach. Many segments of the HDC workflow can be adapted and optimized for the specificities of

medical/epilepsy detection, and we discuss them in this thesis.

1.4.2 Methodological choices for epilepsy detection

Although many studies report impressive levels of accuracy when using ML approaches for epilepsy detection, the widespread adoption of commercial technology for epilepsy monitoring has yet to happen. One reason for the slow progress is that the way studies are designed, algorithms assessed, and results reported is very heterogeneous. It can be difficult to understand the level of evidence these studies provide [250] and also impossible to fairly compare results.

Thus at the beginning of this thesis, in Chapter 2, we discuss methodological choices that have to be made for testing the HDC approach for epilepsy detection, some of which include:

- Data preparation and preprocessing
- Training in a general or subject-specific manner
- Respecting temporal dependencies
- Data segmentation and train-test split
- Evaluation metrics

We also make an overview of epilepsy datasets available and characterize them based on several aspects that can allow engineers to make an informed decision on the appropriateness of the dataset for a specific algorithm, as well as help them and clinicians in the interpretation of the results.

1.4.3 Spatio-temporal data

EEG (iEEG) is the most information-dense biosignal available for epilepsy diagnosis and monitoring. But it is also spatio-temporal, noisy, and non-stationary data, whose efficient encoding to HD vectors poses an essential part of the quality of HD learning. Encoding temporal information is necessary for detecting changes in the signal. Less clear is whether and how to use spatial information between different EEG channels.

In recent years, new approaches have been proposed to encode temporal signals, text, images, and even graphs. But there is a lack of literature on the optimal encoding of spatio-temporal data such as EEG or EMG data. Specifically, for epilepsy detection from EEG or iEEG data, how to encode channel and feature information has not yet been systematically explored in the literature. Thus in Chapter 3 we discuss several questions related to spatio-temporal data:

- Is encoding raw data or data trends to HD vectors enough, or can using carefully designed features improve performance?
- Is there still space for improving performance by designing epilepsy-specific features that capture both amplitude, time, and frequency components?

- What are different approaches to encoding spatial information, and how much can they influence performance?

1.4.4 Unbalanced and large datasets

Class imbalance is one of the serious problems [251] in ML, and is prevalent in medical datasets [252]. Occurrences of epileptic seizures, and therefore epilepsy datasets, are highly imbalanced, i.e., they have a significantly higher portion of inter-ictal (non-seizure) data when compared to ictal (seizure) data [253]. Furthermore, in order to capture enough seizure examples, datasets can be long-term, with several days of recordings per subject.

Dealing with unbalanced datasets is not a rare topic in classical machine learning approaches; however, for HD computing applications, almost no discussion exists in the literature related to this topic. Thus, in Chapter 4 we try to tackle several questions:

- How much is the performance of HD computing algorithms affected when using highly unbalanced datasets? For example, for epilepsy, how representable is performance on a subset of data in comparison to using all available data in the dataset?
- Are there methodological choices for training on highly unbalanced datasets that are more appropriate? What performance metrics should be used to adequately and interpretably assess performance for time-series data with rare events such as epileptic seizures?
- What are the limitations of single-vector-per-class HD models? What happens when data within the same class is highly variable?
- Are there smarter learning approaches that can be designed to deal with highly unbalanced, highly variable, and big data?
- Are there challenges for HDC training using large datasets? Is a coding infrastructure available that enables fast and efficient HD training with large datasets?

1.4.5 Personalized nature

Epileptic seizures are a highly personalized and variable phenomenon, with different individuals experiencing seizures with different frequencies, duration, intensities, and triggers. Thus, most of the HDC models developed by researchers are personalized (only using patient-specific data).

However, for practical applications in wearable devices, general models can play a big role. For example, starting from general models for each new patient and then retraining the model with its individual patterns would be the most efficient approach. Indeed, some researchers also developed general models (using all patients), but the difference between models or how to combine them is rarely discussed.

Thus, in Chapter 5, we want to demonstrate several interesting aspects of HD computing

models when targeting both personalized and generalized models. More specifically:

- We show how HDC models represented as hypervectors can be used to compare interpersonal seizure and non-seizure models.
- We investigate how to create generalized models from personalized models.
- We compare the performance difference between personalized and generalized models.
- We investigate how hybrid models that rely both on personalized and generalized models can be designed.
- Finally, we test knowledge transfer between models of two EEG epilepsy datasets.

1.4.6 Interpretability

The outcome of classical ML approaches is often a "black box" that is not transparent to an expert neurologist, hence cannot be analyzed for understanding the algorithm's predictions or diagnosis, e.g., precisely delineating the ictogenic brain regions.

We believe that the HDC approach has a high potential for interpretable predictions, both in temporal and spatial resolution. Thus in Chapter 6 we discuss several questions:

- Can we design an encoding approach that enables comparison of features and their confidences in time?
- Can this approach also be used to perform feature selection?
- Can, in a similar way, the contribution of individual channels be studied? Can it be further used to perform localization of seizures?
- Is it possible to explain and visualize predictions of the HD model in time per individual features and channels?

2 Experimental methodology for epilepsy detection

As mentioned previously, epilepsy affects a significant portion of the world population, out of which one-third of patients still suffer from seizures despite pharmacological treatments [254]. Thus, there is a clear need for solutions that allow continuous unobstructed monitoring and reliable detection (and ideally prediction) of seizures [27], [28]. Moreover, these solutions will further be instrumental in designing novel treatments, assisting patients in their daily lives, and preventing possible accidents.

However, although many studies report impressive levels of accuracy via machine learning (ML) methods, the widespread adoption of commercial technology is still ahead. The reasons for this are many. Some of them are specificities of epilepsy itself: personalized behavior, highly unpredictable and unbalanced nature, and holistic phenomenon affecting multiple signal modalities, and thus, to get a full picture, multi-modal data is needed from several different sensors. But, another reason for slower progress is that the way studies are designed, algorithms assessed and results reported is very heterogeneous. It can be difficult to understand the level of evidence these studies provide [250] and also impossible to fairly compare results. For example, it is very difficult to compare the performance of various systems when only two quantitative values are reported (e.g., sensitivity and specificity) and when the prior probabilities vary significantly (the a priori probability of a seizure is very low, which means that the assessment of background events dominate the error calculations) [244].

Furthermore, another aspect that makes comparison hard is that large and well-labeled datasets are extremely time-consuming and expensive to produce [73], [255]. Thus, publicly available datasets are of great value, without which progress towards wearable epilepsy monitoring devices would be significantly slowed down. In recent years the number of available epilepsy datasets has increased, but there are important differences in their characteristics. Differences exist in all aspects from recording, filtering, labeling and post-processing, as well as a number of patients and epilepsy types. All of these properties make them more or less appropriate for specific machine learning models or machine learning methods (e.g., personalized or generalized training, seizure prediction, or detection). The biggest number of datasets available are not continuous long-term datasets but a selection of recordings. Testing

algorithms on smaller, non-continuous data might not represent performance in real-life monitoring; thus, directing algorithms development on these datasets might be not optimal.

Thus, in this chapter, we want to, first, present currently available datasets with focus on continuous EEG and iEEG recordings, their main characteristics and also annotation properties that make them interesting for specific types of applications. The characterization of datasets we present and discuss here can allow clinicians and engineers to make informed decisions on interpretation and quality of detection or predictions from a given ML algorithm and whether it is appropriate. Instead of searching for a novel ML algorithms, utilizing and adapting existing ones for more realistic long-term datasets might make new progress in the epilepsy community and make a step forward towards prospective clinical trials [256].

Next we want to bring attention to a number of methodological choices which are usually underreported, but ultimately can have a strong influence on system performance. Such choices are necessarily made during data preparation, training, and also evaluation and reporting of the results. It is important to note that these choices and values of various parameters are not usually detailed in the papers, and without this information the reproducibility of results is also brought into question.

The contributions of this chapter are summarized as follows:

- We present continuous EEG and iEEG epilepsy datasets available for researchers and list a broad range of their characteristics.
- We define several critical characteristics that make them suitable for certain type of models or methodologies. We comment how this can influence further progress in developing wearable devices for epilepsy monitoring.
- We identify a broad range of methodological decisions to be made and reported when training and evaluating the performance of epilepsy detection systems.
- We characterize and assesses the influence of individual choices using a typical ensemble random-forest model and the publicly available CHB-MIT database.
- We provide a broader picture of each decision and give good practice recommendations where possible.

2.1 Epilepsy datasets

Good-quality, open-access, and free EEG datasets are very important catalyst for developing machine (deep) learning algorithms and devices for detecting, predicting and monitoring epilepsy and epileptic seizures [257]. Without them community faces challenges concerning evaluation, standardization, and reproducibility of its studies [26].

The first publicly available EEG dataset was the Bonn EEG time series database [258] published in 2001. It contained ten subjects with the purpose of developing epilepsy detection models. Although it contains only one channel and non-continuous data (100 recordings of 23.6 seconds), it remains a benchmark dataset for many researchers due to its availability. This dataset was later extended as part of the EPILEPSIA project and dataset [26] (but is not any more freely available).

After 2010. several datasets with much bigger volumes and quality of data became available. The quality of the dataset can be measured through various factors such as the number and types of annotations and descriptive information, data structure, the volume of recordings, or also the presence of artifacts and noise. For example, CHB-MIT [242] was published in 2010 but is still commonly used as a benchmark in many papers (cited more than 700 times). It was the first open-source long-term EEG database covering several days of recordings from 23 pediatric subjects. Temple University EEG corpus [259] is the largest publicly available EEG dataset for epilepsy to date. It consists of 667 subjects from semi-long-term recordings. It is an extremely valuable database as it contains seizure-type annotations, labeling of artifacts, as well as, seizure presence in different channels. TUH community also developed various software tools, such as annotation tools, EDF format reading libraries, and toolboxes for seizure detection.

Over the years, many datasets have been made available and are listed in two recent reviews on epilepsy EEG datasets [256], [257]. Moreover, there have been several online competitions organized: the Kaggle Upenn and Mayo Clinic's seizure detection challenge [260] or, for example, recent BIOMED Seizure Detection Challenge [261]. These competitions are helping in comparing algorithms in a systematic way and are a great way forward. However, the datasets and scope of competitions were still quite focused and limited. Competitions are motivating but are not necessary. More important are all-time-actual platforms where it is possible to compare the performance of new algorithms as being developed in a fair way. Here the good example is the Epilepsy Ecosystem initiative and their 'My Seizure Gauge Data' [262] project focusing on long-term datasets from wearable devices.

Furthermore, many public datasets contain only a few subjects (e.g., less than 10) [258], [263], [264]. Since a small number of patients significantly limits the utility of the dataset (e.g.it is hard to develop generalized models on the small number of patients), we excluded those datasets from the list. Furthermore, many datasets contain only short-term recordings or recordings where samples are shuffled in time, so the time component is lost [260], [264]–[268].

Table 2.1: Continuous data EEG epilepsy datasets. Both long-term and short-term datasets are listed. Only datasets with more than 10 subjects are listed.

Dataset	Availability	Link	Reference	Year	Population
CHB-MIT	freely available	physionet.org/content/chbmit	[242]	2010	Pediatric
SWEC-ETHZ	freely available	iEEG-swez.ethz.ch	[269]	2018	Adult
EPILEPSIAE	payed	epilepsy-database.eu	[26]	2012	Adult
TUH EEG seizure (TUES)	free, but registration needed	isip.piconepress.com/projects/tuh_eeg/downloads/tuh_eeg_seizure	[259]	2017	Adult
Helsinki University Hospital	freely available	zenodo.org/record/1280684#.YQJpgI4zbIU	[270]	2018	Pediatric (neonates)
HFO epilepsy dataset	freely available	openneuro.org/datasets/ds003555/versions/1.0.1	[271]	2021	Adut and paediatric
Repomse	private	-	[272]	2019	Adult

In Table 2.1, we list continuous EEG (or iEEG) datasets that contain either long-term or short-term (less than 24h of recordings per subject) recordings. More specifically, we list only datasets that contain at least ten subjects, as it is hard to test the generalization of models developed on a small number of patients. In fact, there are more datasets, but some of them are not well documented or are not freely available and thus are not listed here.

2.1.1 Dataset characteristics

There are huge variations across several characteristics of datasets. The difference in acquired channels and their exact locations, the difference in sampling frequency, and preprocessing of datasets are just some of them. For continuous datasets from Table 2.1, in Table 2.2, many characteristics of recording are listed: the number of subjects, channels recorded, if there were other modalities than EEG (or iEEG), sampling frequency, length of recordings and how big is the database in total. After the recording stage, the difference in the preparation of the data, what data is excluded (artifacts, pre-ictal data etc.), and whether the full long-term data or subsets with more balanced scenarios are provided is another big difference.

Even if these characteristics don't have to be identical (or will not be possible to be identical),

this significantly limits the comparison of the results between different datasets. Furthermore, it also limits the possibility of merging datasets to have bigger datasets to train or study knowledge transfer between datasets. Knowledge transfer between models of different datasets is still not yet a well-researched topic, which highly depends on available datasets and their characteristics.

One thing to mention is that despite datasets being made available, documentation is often lacking. For example, for many datasets, overall recorded seizure duration, or distribution of the number of seizures per patient, is unknown and must be calculated manually. This can be quite challenging due to significant differences in dataset formats and processes needed to read and use datasets. Many datasets now use *.EDF* file format to store recordings, but a lack of standardization in the datasets is still present.

The characterization of datasets we present and discuss here can allow clinicians and engineers to make an informed decision on the interpretation and quality of detection (or predictions) from a given ML algorithm and whether it is appropriate. Instead of searching for novel ML algorithms, utilizing and adapting existing ones for more realistic long-term continuous datasets might make new progress in the epilepsy community and towards more clinical trials [256].

In the following subsections, we divide the characteristics of datasets into five important categories: number of patients and volume of the dataset, length of recordings, signal modalities available, channels recorded, and annotations available.

Table 2.2: Continuous data EEG epilepsy datasets and their main characteristics.

Dataset	Number of subjects	Number of channels	Sampling frequency	Primary modality	Other modalities	Length of recordings	Total size	Data format
Long-term continuous datasets								
CHB-MIT	23	23 to 26	256 Hz	EEG	vagal nerve	long-term	40 GB	.edf
SWEC-ETHZ long-term	18	>50	512 or 1024 Hz	iEEG	no	long-term	1.53 TB	.mat
EPILEPSIAE	275	>18	250 Hz to 2.5 kHZ	EEG 217 subj, iEEG 58 subj	ECG, for 65% also EMG	long-term	240 GB	proprietary
Shorter continuous datasets								
TUH EEG seizure (TUES)	667	20 to 32	250, 256, 400 and 512 Hz	EEG	ECG and photic stimulation	1 hour pruned segments	54 GB	.edf
Helsinki University Hospital	79, but 39 with seizures	19	256 Hz	EEG	ECG and respo-ratory effort	60 to 90 min/-subj	4.3 GB	.edf
HFO epilepsy dataset	30	52	2000 Hz	EEG	no	3h at the begin-ning of a sleep /subj	15 GB	.edf
SWEC-ETHZ short-term	16	>50	512 or 1024 Hz	iEEG	no	3 min preictal, ictal, 3 min postictal	11 GB	.mat
Repomse	869	>20	256, 512, 1024 Hz	EEG and some iEEG	ECG, SPo2	1 min seiz and 1 min non-seiz	242 GB	.mat

Number of patients and volume of dataset

Since epilepsy exhibits highly personalized patterns, the number of patients is an important factor in testing how well models perform on a range of patients. In the case of developing generalized models, the number of patients (and, in general, the volume of the dataset) has to be big enough to enable the good generalization of the models. The model can generalize well on the trained group, but if the number of seizures/patients is small, it will most likely fail to capture important characteristics for patients outside the trained group. The range of patients in public datasets is very broad. First datasets, such as the Bonn dataset [258], contained only a few subjects, while today, we have a public TUH database that contains more than 600 patients, and probably more private databases. For example, a private database used in this thesis, Repomse dataset [272], contains over 800 patients.

However, large datasets are also challenging. Very often, there is less consistency between recordings due to the practical complexity of recordings. For example, the number of channels or their locations might not be identical between all the subjects [26], [259], there might be differences in lengths of recordings, etc. From a training perspective, large datasets are much more demanding to train and often require significant compute power, usage of servers, and GPUs.

Length of recordings

Datasets can be long-term, meaning that they contain continuous recording over several hours or even days. They are long-term even if there are short-term gaps in data due to disturbances in recordings (e.g., electrodes lost contact, the patient had to go to the toilet, etc.). Good examples of such databases are CHB-MIT, SWEC-ETHZ, EPILEPSIAE.

Other options are shorter data subsets, usually created from longer recordings, containing seizure data, and then some relatively short non-seizure recordings selected to satisfy certain criteria. For example, SWEC-ETHZ consortium also provided SWEC-ETHZ short dataset, where they selected 3 min of non-seizure data before seizure data, which is again followed by 3 min of non-seizure data. Similarly, in Repomse dataset, a selection of 1 min seizure and 1 min of non-seizure data is created.

Short data subsets are often a more practical and faster solution to developing algorithms and train ML models. On the other hand, they contain only a small selection of non-seizure data that might not be representative of the whole spectrum of non-seizure periods (e.g., sleeping, moving, eating, mental effort, etc.) that occur regularly in everyday life. Further, they do not represent inherent disbalance in seizure and non-seizure data, which is one of the main characteristics of epilepsy. Short datasets do not represent the unpredictable nature of epileptic seizures and as such, might not necessarily model and perform well on wearables used in everyday life. For example, it has been shown that seizure likelihood increases if there was a recent seizure. Also, in a certain group of patients, it has been found that cyclic trends

such as the timing of the day can be correlated with seizure occurrences [273]–[275]. These patterns can be only studied from long-term databases.

Having long-term recordings enables studying not only the detection of seizures but also the prediction of seizures. Moreover, different prediction parameters can be tested, for example, how early seizures can be predicted, and how multiple seizures interact and influence the probability of future seizures. Long-term recordings also enable an analysis of post-ictal patterns.

Modalities recorded

Most of the epilepsy datasets available are EEG datasets, as EEG (or iEEG) is the most informative physiological signal we have to monitor epilepsy. Yet, as epilepsy is a complex disease that translates to changes in several physiological signals, some datasets contain other modalities such as an electrocardiogram (ECG), electromyogram (EMG), or even photoplethysmography (PPG), respiration (RPS), electrodermal activity (EDA), accelerometry (ACC) and skin-temperature (SKT).

Even though most of the work for detecting epilepsy is based on EEG data (as well as this thesis), more and more effort is put into detecting seizures from modalities that are more appropriate for outpatient and wearable monitoring (ECG, PPG, EDA, ACC, SKT, RSP) [262].

For example, EMG-based seizure devices attached to, e.g., arms, seem to be able to detect generalized tonic-clonic seizures (GTCS) with sensitivities of more than 90% [90], [168]. Similarly, studies have also shown that there are significant changes in the heart rate (i.e., tachycardia or heart rate variability) [173], [174] that can also add an additional set of information and improve detection and prediction algorithms.

Channels recorded

EEG databases can contain scalp channels that usually follow international 10-20 EEG electrode placement standards. If intracranial EEG is recorded, it can follow approximately 10-20 placement with additional electrodes usually localized around the epileptogenic zone. In some datasets, the selection of channels is already made during preprocessing of the dataset, and only some channels are made available for researchers. One thing to note is that even if datasets follow the same placement, the naming of each channel can be different. Furthermore, in some cases, unipolar recordings are saved (e.g., Repomse), while in some cases, differential recordings are saved (e.g., CHB-MIT). As it is easier to go from unipolar recording, so bipolar, unipolar recordings are recommended.

The number of channels and location of channels in the dataset is an important characteristic of the dataset. For example, having datasets with the same distribution of channels can enable combining datasets or comparing performance between different datasets. Furthermore,

training on one dataset and testing on different datasets is only possible if (at least some) channels are located on approximately the same location. If a different number of channels is used, this also usually means a different number of features and sizes of models and thus makes testing the knowledge transfer between models of different datasets impossible.

Even if final wearable devices might only utilize a small selection of channels to be practical and inconspicuous, having more channels recorded originally can be very useful. As mentioned above, it can enable comparison between datasets, but also, with more channels available, they can be used for knowledge distillation for reduction of a number of channels [276]. Knowledge distillation is the process of creating better labels (e.g., probabilities instead of bipolar labels) from more complex models (e.g., containing more channels) and then improving the quality of smaller (or simpler, e.g., having fewer features or needing fewer channels) models by having more discriminative labels.

Intracranial or scalp recordings with many channels are interesting for observing seizure onsets and spread through the scalp [73]. Detecting affected brain lobe(s) or rating channel importance could, when provided to a neurologist or a neurosurgeon, help determine epilepsy seizure types [140].

Table 2.3: Continuous data EEG epilepsy datasets and their annotations.

Dataset	Seizure annotations	End of seizures	Additional annotations	Patient information	Pers/Gen models	Prediction vs. detection	Seiz. type classification
Long-term continuous datasets							
CHB-MIT	S/NS	marked	no	age, gender	both	both	no
SWEC-ETHZ long-term	S/NS	marked	no	no	both	both	no
EPILEPSIAE	S/NS, seizure type	marked for clinical and EEG seizure	seizure location (front/temp/ else)	age, gender, epilepsy character-istics, medications	both	both	yes
Shorter continuous datasets							
TUH EEG seizure (TUES)	S/NS, seizure type	marked	artifacts and channels	age, gender, clinical history, medications	both	both	yes
Helsinki University Hospital	S/NS	no	annotations of 3 experts	no	both	both	no
HFO epilepsy dataset	HFO labels	NA	deep sleep stage annotations	age, gender, clinical diagnosis	both	both	NA
SWEC-ETHZ short-term	S/NS	marked	no	no	both	detection, limited prediction	no
Repomse	S/NS	no	no	no	both	detection, limited prediction	no

Annotations available

Lastly, but extremely important is whether annotations are available. Neurologists usually do annotations while observing patient video recordings. The minimal annotation that datasets need to contain is the start of the seizure. Annotation of the end of the seizure episode is more complex than the start, as seizures often exhibit post-ictal behavior that can be very similar to ictal data. However, annotating the end of a seizure is extremely helpful, as the duration of the seizure can vary significantly and training on potentially wrongly labeled data can significantly complicate training.

Next, most datasets only contain binary labeling; seizure vs. non-seizure (ictal vs. non-ictal or inter-ictal data). Some datasets also include information about seizure type (e.g., focal (simple and complex partial) or generalized (GTSC) seizure as in EPILEPSIAE). These annotations are rare but can be very useful. More specifically, generalized seizures are much easier to detect, and thus unique training approaches for focal seizures might be needed in the future. Furthermore, since some seizures can be very local, rather than spreading to the entire brain, information on the channels where epileptiform patterns are most visible can also be very helpful, but it is rarely annotated.

In the end, EEG data is often compromised with various artifacts (e.g., eye blinking, muscle artifacts, etc.). Many artifacts are also commonly present during the seizures, making it harder to distinguish ictal patterns. Attempts have been made to detect artifacts before or in combination with seizure detection [277], [278].

2.1.2 Datasets used in this thesis

In this thesis, we utilize several epilepsy datasets, which we describe in detail below.

CHB-MIT

The CHB-MIT epilepsy dataset is a widely used open-source dataset for epilepsy detection [242]. It is a good representative of continuous, long-term monitoring (over several days) database. CHB-MIT is a scalp EEG database, with a total of 982.9 hours of data recorded at 256Hz. It consists of 183 seizures forming in total 3.2 hours or 0.32% of labeled ictal data, from 24 subjects (5 males and 18 females, one repeated patient) with medically-resistant seizures ranging in age from 1.5 to 22 years (10 ± 5.7 years old). On average, it has 7.6 ± 5.8 seizures per subject, with an average seizure length of 58.6 ± 65.0 s.

Data are recorded using the bipolar montage (10-20 system [279]) at a sampling frequency of 256 *Hz* and using 16-bit resolution. Originally it contained 18 to 24 channels, but for the sake of consistency, we use the 18 channels that are common to all patients (i.e., FP1-F7, F7-T7, T7-P7, P7-O1, FP1-F3, F3-C3, C3-P3, P3-O1, FP2-F4, F4-C4, C4-P4, P4-O2, FP2-F8, F8-T8, T8-P8, P8-O2, FZ-CZ, CZ-PZ). The data set is generally organized as one hour-long files, in the

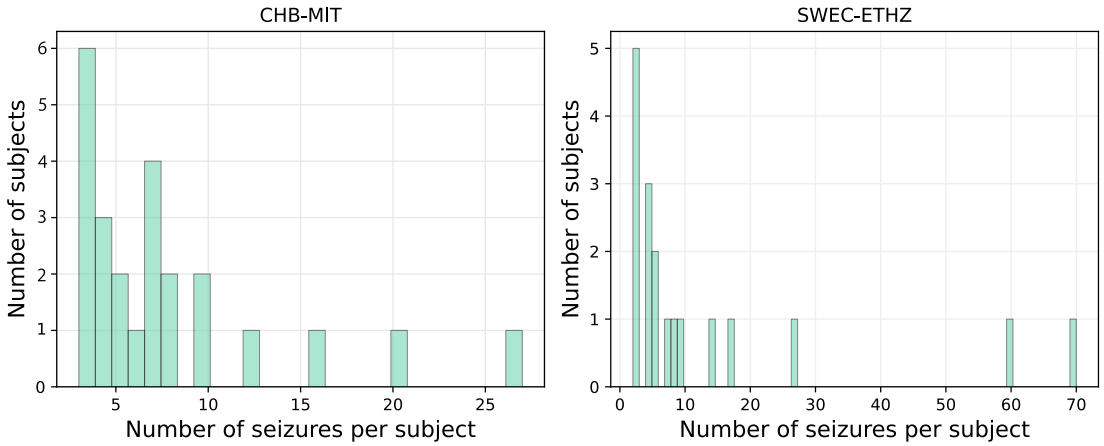

Figure 2.1: Histograms of number of seizures per subject for CHB-MIT and SWEC-ETHZ dataset.

European data format (*.edf*). However, some subjects have files of two or four hours long, and some files containing ictal data are less than one hour long. Furthermore, subject *chb12* has few files with data acquired on a monopolar montage, which we excluded to keep a similar methodology among all subjects.

SWEC-ETHZ

The SWEC-ETHZ dataset [269] presents 2656 hours of long-term iEEG data from 18 patients, containing a total of 116 annotated seizures. It was created by Inselspital Bern epilepsy surgery program. Data are recorded in a monopolar montage at sampling rates of 512 or 1024 *Hz*, median-referenced, and band-pass filtered between 0.5 and 120 *Hz* (zero-phase, 4^{th}-order Butterworth filter). The data are organized as 1 *hour* long *.mat* files accompanied by one extra file per subject, including seizure annotations.

SWEC-ETHZ also contains a shorter database created from a subset of patients from the original database described above. The shorter version contains 16 patients and, in total, 100 seizures, with an average of 6.3 ±3.8 seizures per subject. Each recording consists of a 3-minute interictal segment, an ictal segment (ranging from 10 to 1002 seconds), and finally, a 3-minute interictal segment. The number of channels was variable per patient due to differences in implanted iEEG electrodes.

Repomse

Repomse database [272] is an epilepsy database curated by the Lausanne University Hospital. Originally, the database contained EEG recordings of 869 patients with an average of 3.3 seizures per patient. A more detailed distribution of the number of seizures per subject is shown in Fig. 2.2.

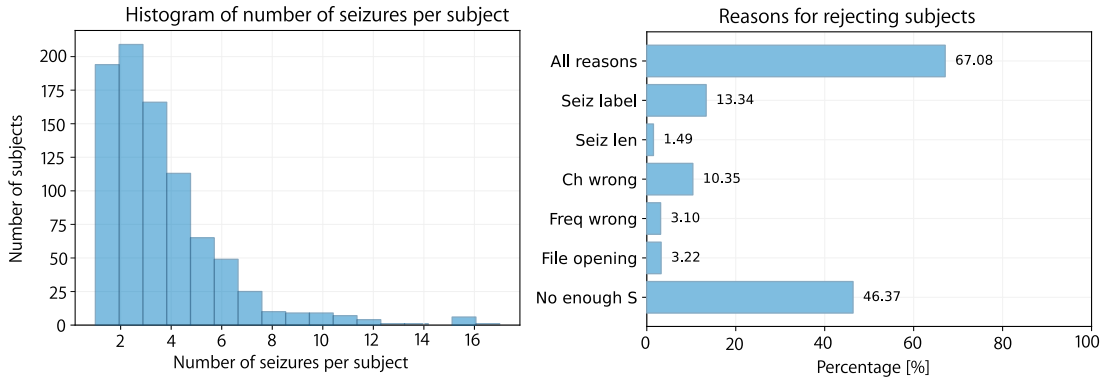

Figure 2.2: Statistics of the Repomse dataset: number of seizures per patient in original database, and also reasons for rejecting specific subjects. In the end we keep 286 subjects which satisfy our criteria.

We curated the dataset by selecting only patients with at least three seizures (to be able to create personalized models), with identical 18 channels (i.e., FP1-F7, F7-T7, T7-P7, P7-O1, FP1-F3, F3-C3, C3-P3, P3-O1, FP2-F4, F4-C4, C4-P4, P4-O2, FP2-F8, F8-T8, T8-P8, P8-O2, FZ-CZ, CZ-PZ) and with a sampling frequency of at least 256Hz. If the sampling frequency was 512 or 1024Hz, we downsampled it to 256Hz to have it identical for all subjects. In Fig. 2.2, reasons for rejecting specific subjects and the percentage of subjects affected by it are shown as well. Finally, we kept 286 subjects, which is still a much larger number of subjects than we could have had from any other publicly available dataset and, as such, was ideal for studying generalized models.

Original recordings contain a single seizure with a maximal length of 1 min, with usually 3 minutes of interictal recording before the annotated seizure onset. Similarly, as in [280] we exclude the pre-ictal period in the duration of 1 min, and utilize 1 min of inter-ictal data. In our case, we split it into half and put around 1 min of seizure data. This way, every file contains a balanced amount of seizure and non-seizure data.

2.2 EEG features

The primary focus of this thesis is not on designing novel features for epilepsy detection. In fact, in the literature, there are many works that define and use a broad set of EEG features.

Generally, the most commonly used EEG features can be divided into time-domain, frequency-domain, and nonlinear features.

The first category is based on characterizing the time information of the EEG signals. Here, the typical features would be the mean or median amplitude and variance or standard deviation of the signal in a given window of EEG data. However, more complex statistical features, such as skewness, kurtosis, Hjorth parameters [281], zero crossing number, line length, etc., can be measured [140]. In this thesis, we will also propose one new time domain feature, called approximate zero-crossing (AZC), in Chapter 3.3.

The frequency domain features are then very informative when dealing with EEG data. Specifically, there are several physiologically chosen frequency bands: delta [0.5, 4] Hz, theta [4, 8] Hz, alpha [8, 12] Hz, beta [13, 30] Hz, and gamma [30, 45] Hz. For example, alpha waves are usually associated with being relaxed and/or with closed eyes. On the other hand, beta waves are more indicative of focused active thinking, often also when being anxious. Theta waves, for example, are associated with drowsiness. Epileptic seizures also affect the distribution of the power of the EEG signal in different frequency bands [282], [283]. Frequency features are usually extracted after performing the FFT or wavelet decomposition analysis, and can be expressed as the absolute power of a signal in those frequency bands or relative when compared to the total power of the signal. In fact, relative powers of certain frequency bands are the most widely used frequency domain features in all fields of EEG signal analysis [284]. More complex features, but the energy, or the previously mentioned statistical features, can also be calculated from the frequency spectrum [284]. Methods like Short-time Fourier transform (STFT) or Wavelet Transform Decomposition (WT) allow us to extract time-frequency features. For example, phase coupling of different frequency components can be obtained with higher-order spectral analysis.

Finally, there have been various ideas to characterize EEG dynamics that are based on nonlinear approaches: fractal analysis extracting multiple features such as Lyapunov exponents [285], Lempel-Ziv complexity [286], central tendency measures [287], auto-mutual information [288] and others. Ultimately, entropies are used to measure uncertainty or randomness in the observed time series. Shannon was the first to propose what we call today Shannon entropy [289]. However, today there are many more entropy measures such as Renyi [290], Tsallis [291], approximate [292], sample [293], permutation entropy [294] and more.

Finally, there are also approaches to characterize the spatio-temporal relationship between EEG channels, such as mutual information [295], coherence [296] or Granger causality [297].

Based on lab experience in EEG signal processing for epilepsy, we have selected a smaller selec-

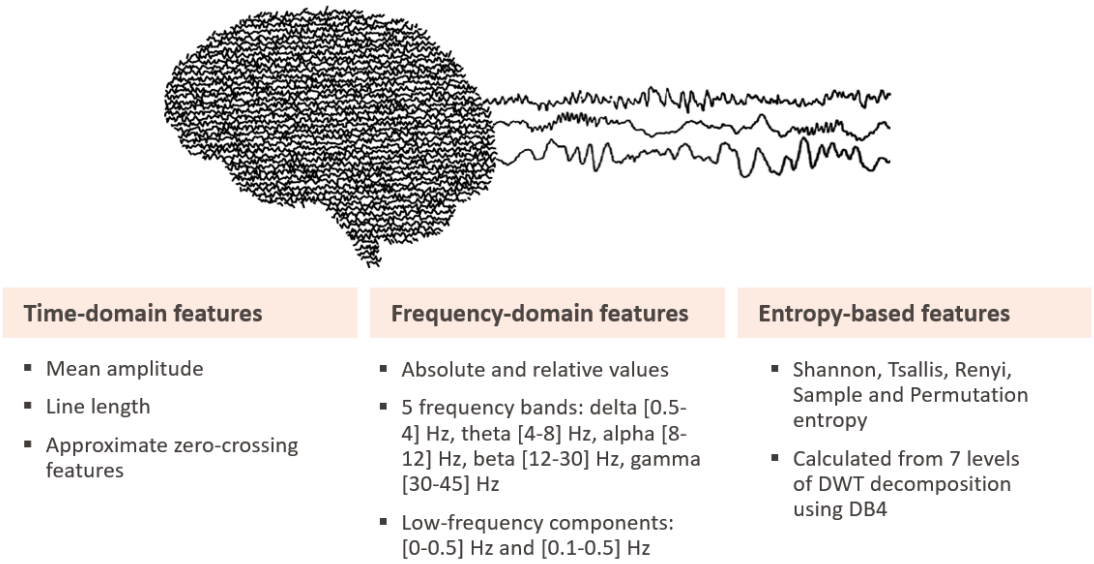

Figure 2.3: Three types of EEG features that are commonly extracted and used for seizure detection are illustrated. Mentioned are only features used in the scope of this thesis. In literature, more features are available.

tion of features that are discriminative but also relatively simple enough to be implemented on wearable devices. In Fig. 2.3, we listed and grouped the features used in the scope of this thesis. We focus on few simple time domain features, several basic frequency domain features and entropy features from the time-frequency decomposition of the signal as presented in [298]. In different parts of the thesis, a selection of features is further made to lower the number of features even more.

2.3 Methodological choices

There are many methodological choices to make when evaluating machine learning algorithms and systems in terms of their performance and further suitability for real-life applications. These choices, even though sometimes seeming small and irrelevant, can significantly impact the performance and repeatability of such results in practice. Yet, we often overlook some of them, go with the most common ones, or forget about the possibility of testing them. In the this section we go through the most important choices, discussing data preparation, training and testing methodology, and performance measures, as listed in Table 2.4. We will later show how they influence epileptic seizure detection.

2.3.1 Data preparation

An important part of evaluating machine learning algorithms is the data used to train and test the algorithm. A well-known practice is that training, validation, and test subsets must be chosen without overlap and be statistically independent to avoid the effect known as 'data leakage'. But the question that is less discussed, is how representative is the data that we use. With the increasing amount of big-data collected using IoT and wearable devices, big datasets are no longer rare. Such big datasets are incredibly valuable and essential for having more ML/AI-powered devices in everyday life, but they also bring certain challenges. Training on such a huge amount of data, especially for computationally demanding or memory intensive algorithms, or without lots of computational resources, can be complex, slow, and even potentially not feasible. For this reason, a common approach is to create smaller subsets of available datasets.

In the case of epilepsy, it is characterized by recurrent but unpredictable electrical discharges in the brain. Epilepsy episodes can last from just a few seconds to a few minutes. Overall, when looking at the recorded data, the percentage of seizure data is extremely small, commonly less than 0.5%. In Table 2.5 we show characterization, including the percentage of seizure data, for two publicly available datasets that contain completely continuous recordings recorded from several days (rather than preselected subselection of data, that is common in other datasets). This huge imbalance in epilepsy recordings leads to the common choice of creating a data subset that contains all seizure signals but only a reduced amount of non-seizure signals. This step of creating smaller or even balanced data sets simplifies training, makes reporting performance clearer, and speeds up the research process. Most of the articles that tackle the problem of epilepsy detection do not use whole long-term epilepsy recordings, but rather subsets of data and also very rarely discuss the influence of this decision on their results.

In this section, we address this question by testing several epilepsy subsets created from a main dataset. We evaluate the influence of using all or only some data samples, as well as the impact of the seizure to non-seizure imbalance ratio. We also test the influence of data splitting during the training and cross-validation folds, and show that this choice can be very critical and make a big difference in whether the proposed algorithm will work in practice

Table 2.4: Overview of all methodological choices tested

Data used	Subsets of data with different imbalance ratios (e.g. Factor1 and Factor10)
	All data, with different splits into training folds (SeizureToSeizure, 1h/4h windows)
Training	Cross-validation type: Leave-one-out (L1O) or Time-series-CV (TSCV)
	Window step: 0.5 to 4s, with 4s windows
	Personalized models or generalized models
Performance metric	Episode and duration-level performance
	Micro or macro CV folds averaging

when all data is used, without the possibility to perform any selection.

2.3.2 Generalized vs. personalized models

In many applications where underlying data patterns are highly specific, such as in many biomedical use-cases, there are two approaches to training; personalized and generalized training and models. Epilepsy is a good example of this, where underlying electroencephalography (EEG) patterns are highly variable between EEG channels, recording sessions, and subjects. Personalized training means that data from the same subject is used to train the model. This leads to as many ML models as subjects we have. Generalized training, on the other hand, would lead to one ML model for all subjects. To avoid data leakage (and enable comparison with personalized models), every subject has its own generalized model trained on all subject's data but that test subject, which is also known as the leave-one-subject-out approach.

On one hand, personalized modes can capture subject-specific patterns better, but are also trained on less data in total, which can sometimes be limiting, as some subjects have very few seizures recorded. On the other hand, generalized models are more complex to train as they are trained on more data and can also be less subject-specific but may be more interesting for building large-scale wearable outpatient systems. For widely accessible wearable devices, it will not be realistic to record each subject's data to create personalized models, but generalized models will have to be used. This is also necessary for subjects that have very rare occurrences of seizures, which cannot be captured in a few days of in-hospital recordings.

2.3.3 Respecting temporal data dependencies

Another aspect of data that is commonly forgotten is that all data is recorded in time, and that sometimes this imposes some unavoidable statistical dependencies. Some underlying patterns that our ML algorithms can use, can only exist in a certain order, and for this reason it might not be fair to use data that is in the future to train and then test using data that was

Table 2.5: Characterization of two publicly available long-term epilepsy databases (NrSbj: number of subjects, NrSeiz: number of seizures, DatLen: dataset length [h], TotSeizLen: total seizure duration [h], PercSeizLen: percentage of seizure duration [%])

Dataset	NrSbj	NrSeiz	DatLen	TotSeizLen	PercSeizLen
CHB-MIT	24	198	982.9h	3.2h	0.32 %
SWEC-ETHZ	18	116	2656h	3.6h	0.13 %

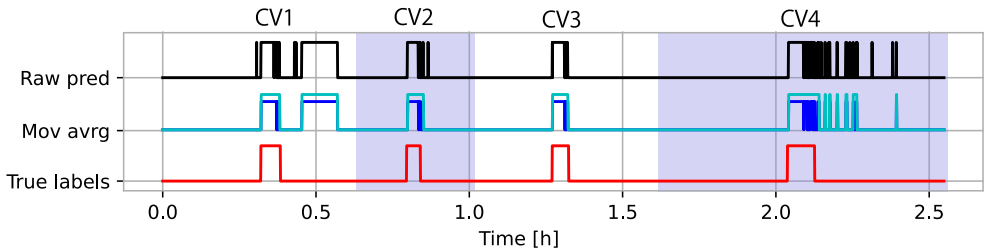

Figure 2.4: Epilepsy model predictions example. Predictions without any postprocessing and with two types of postprocessing are shown as well as true labels. Of interest are the distributions of false positives. The distributions of false positives are of particular interest.

before it. On one hand, it can miss some useful patterns for detection, and on the other hand, it can lead to potentially unfeasible results for in-practice applications.

In this case, two parts of the ML workflow have to be considered. Often, data samples are shuffled before training, whereas for temporal data, this might not be advisable. Furthermore, if some statistical knowledge on the distribution and length of certain classes is available, this knowledge can be used to post-process predicted labels and lower misclassification chances. For example, in the case of epileptic seizures, it would not be realistic that an individual suffers an epileptic seizure of 1-second duration every minute.

Secondly, the temporal aspect of data is relevant when choosing the cross-validation (CV) approach. A common CV approach [242], [299], [300], for personalized training is leave-one-seizure-out, a subtype of leave-one-out CV (LOOCV) approach. In this approach data from one seizure is left out for testing, and seizures that come before but also after will be used for training. This is unfortunately non-realistic approach in which future data is used to evaluate the current test set.

On the other hand, in the time-series-cross-validation (TSCV) approach [301], [302] only previously acquired data can be used for training. This means that if files are ordered in time, for the first CV fold only one file will be used for training and the one after it for testing. For the following CV folds, as illustrated in Fig. 2.5, one more file is always added to the training set (the file previously used for testing) and testing is done on the next available file. This CV approach is rarely used in the literature, but is the only feasible approach for online data training (and inference) on wearable devices.

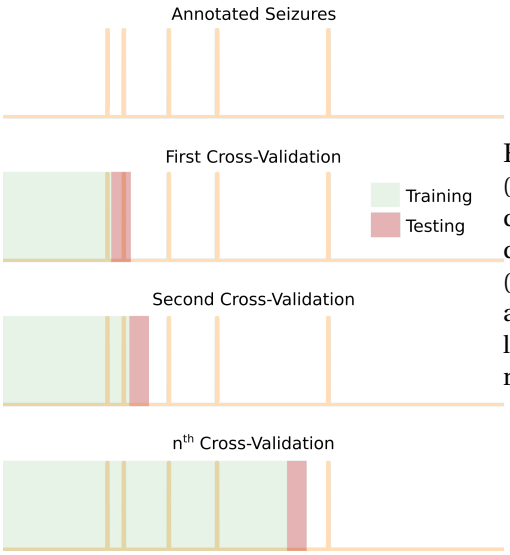

Figure 2.5: Proposed time-series cross-validation (TCSV) approach. This approach preserves the temporal dependency among data, i.e., only previously recorded data is used for training, and the next data segment (e.g., one file) is used for testing. All test predictions are then concatenated and compared with the original labels. This approach represents the assessment of a real-life application more appropriately.

2.3.4 Data segmentation

Typically, features are extracted from fixed-size windows of data, and calculated with 'moving-window' repeatedly in shifts of a chosen step size. Here, two parameters have to be decided: window size (WS) and window step size (WSS) for which we move the feature extraction window. Choosing a larger window size might be necessary when extracting frequency information, but it also limits the possibility of detecting very short patterns. Similarly, a smaller step size can decrease detection latency but increases the computational costs of the algorithm due to more frequent feature extraction. These parameters can be optimized according to several aspects: features used and their properties and complexity, latency requirements, or available computational resources. If none of these are limiting factors, parameters are generally optimized in terms of performance. It is interesting to notice how much performance can change depending on these choices. More importantly, parameter choice and the reasoning behind it should be mentioned and documented in papers.

2.3.5 Evaluation metrics

For temporal and sequential data, standard performance evaluation metrics such as sensitivity and specificity might not always be the most appropriate ones and can even be misleading [303]. Evaluation metrics must ultimately reflect the needs of users and also be sufficiently sensitive to guide algorithm development [244]. As Shah et al. stated in [245] there is a lack of standardization in evaluating sequential decoding systems in the bioengineering community.

The same authors compare five popular scoring metrics for sequential data in [244]. Among them, the most interesting are 'Epoch-based sampling' (EPOCH), 'Any-overlap' (OVLP), and 'Time-aligned event scoring' (TAES). EPOCH treats the reference and hypothesis as temporal signals, samples them at a fixed epoch duration, and counts errors (TP, TN, FP, and FN)

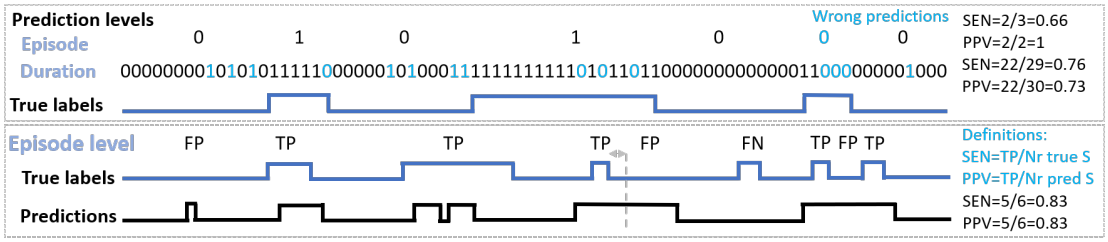

Figure 2.6: Illustration of duration and episode-based performance metrics.

accordingly. For an epoch duration of one sample, this metric processes data sample-by-sample and results in typical performance measures (such as accuracy, sensitivity, specificity, F1 score etc). The OVLP measure [304], interprets signals as a series of same-label episodes and then assesses the overlap in time between reference and the hypothesis. It counts a 'hit' in case there is any overlap between the reference and hypothesis. In Fig. 2.6 we illustrate several use cases, how errors are counted and what is the final performance measure. Authors in [244] also propose the TAES metric which combines EPOCH and OVLP information into one metric. The approach is very similar to OVLP, but rather than simply considering if there is any overlap between reference and hypothesis episodes, the percentage of overlap is measured and weighs the errors (TP, TN, FP, FN) accordingly. Here we want to demonstrate the difference in performance in the use-case of epilepsy detection, depending on the chosen performance measure, and also how these performance metrics can be used to interpret the quality of algorithm predictions.

In this thesis, we use two metrics to measure performance.

- **Duration level performance**: It is a standard sample-by-sample based performance where each sample has to be correctly classified to have 100% accuracy. Moreover, it is the same as the EPOCH approach described above, with an epoch duration of one sample. TP, TN, FP, FN are detected sample-by-sample, which are further used to calculate sensitivity (TPR), precision (PPV), and F1 score. We call this duration-based performance which characterizes how well are seizures detected with regard to their full length.
- **Episode level performance**: This metric cares only about whether each seizure episode has been detected, not caring exactly about the predicted seizure duration. It corresponds to the OVLP metric described above, which detects overlaps between predicted (hypothetic) and reference seizure or non-seizure episodes. These metric is very easily interpretable. For example, if sensitivity on episode-level is 80% and there were 10 seizures, it means that 8 episodes were detected, but 2 were missed.

Metrics on these two levels give us a better insight into the operation of the proposed algorithms. Furthermore, the performance measure often depends on the intended application and plays a big role in the acceptance of the proposed technology. The TAES metric proposed

in [244] is an interesting approach to combine both metrics but is harder to interpret and thus it is not used here.

For both episode and duration level performance we further measure:

- **Sensitivity**: It is also called true positive rate (TPR) or recall and it is defined by the following equation:

$$TPR = \frac{TP}{AllP} = \frac{TP}{TP + FN} \tag{2.1}$$

- **Precision**: It is also called positive predictive value (PPV) and is defined by the following equation:

$$PPV = \frac{TP}{PredictedP} = \frac{TP}{TP + FP} \tag{2.2}$$

- **F1 score**: It is the harmonic mean of precision and sensitivity, defined by the following equation:

$$F1Score = \frac{2 * TPR * PPV}{TPR + PPV} \tag{2.3}$$

Finally, in order to have a single measure for easier comparison of methods, we calculate the geometric mean value of F1 score for episodes ($F1E$) and duration($F1D$) as defined by the following equation:

$$F1DEgmean = \sqrt{F1D * F1E} \tag{2.4}$$

Another performance measure with a strong practical impact, and thus often used for epilepsy detection is the 'false alarm rate' (FAR), or the number of false positives per hour/day. Clinicians and patients see this measure as more meaningful than many more commonly used metrics, and are very demanding in terms of performance, requiring it to be as low as possible for potential wearable applications (e.g., less than 1 FP/day) [245]. This necessitates exceptionally high constraints on the required precision too (usually much higher than 99%).

Finally, to quantify global performance, the accumulated performance of all cross-validations folds has to be calculated. But here are also choices to be made. One can measure the average performance of all CV folds (micro-averaging) or can, for example, append predictions of all test files next to each other and only then measure performance on all appended data (macro-averaging). In Fig. 2.4, an example of predictions (also with moving-average postprocessing) for all files of one subject is given. What is important to notice is the distribution of false positives in time and over the different files/CV folds. Most commonly, false positives occur around the seizures. However, there can potentially be a fold(s) with an unexpectedly large number of false positives. If the final performance is measured as an average of the performances of each fold, a fold with many false positives, as in Fig. 2.4, will have a lower influence

on the total performance than if all predictions are appended and performance is measured only after. This potential overestimation of performance when averaging cross-validations should be also taken into account.

2.4 Experimental setup

In this work, we use the CHB-MIT epilepsy dataset, a widely used open-source dataset for epilepsy detection [242] (described in Sec. 2.1.2), as it is a good representative of continuous, relatively long-term EEG monitoring.

We extract 19 features from each of the 18 channels, similar to [305], calculating them in 4-second windows with a moving step of 0.5 seconds (unless otherwise specified). We use two time-domain features, mean amplitude and line length, and 17 frequency domain features. Both relative and absolute values of power spectral density in the five common brain wave frequency bands are used: delta: [0.5-4] Hz, theta: [4-8] Hz, alpha: [8-12] Hz, beta: [12-30] Hz, gamma: [30-45] Hz, and low-frequency components: [0-0.5] Hz and [0.1-0.5] Hz. Before extracting the features, the data is filtered with a 4th-order, zero-phase Butterworth bandpass filter between [1, 20] Hz.

As an algorithm to test the range of parameters mentioned, we choose a highly popular but also feasible algorithm for portable and outpatient monitoring devices. We implemented a random forest classification algorithm, which is based on an ensemble of 100 decision trees to reduce the model overfitting. It is fast and lightweight, both in model size and memory footprint [306], and it has been extensively used for EEG-based seizure classification [36], [140], [241]. In the end we postprocess predicted labels with a moving average window of 5 seconds, and majority voting to smooth predictions and remove unrealistically small seizures. If seizures are closer than 30s, we merge them into one.

2.5 Results

2.5.1 Evaluation metrics

Fig. 2.7 shows the average performance of personalized models for all 24 subjects from the balanced CHB-MIT dataset. For both the episode- and the duration-based performance, the sensitivity, precision, and F1 scores are shown, as well as an accumulative measure, the mean of the F1 score for the episode and duration level ($F1\_DE$). Episode sensitivity is on average 100%, meaning that except for a few cases, all seizure episodes were detected. Looking at duration level sensitivity, it is clear that even if seizure episodes were perfectly detected, their whole duration was not always detected. Looking at the precision, it is clear that there are also false positive predictions, and more of them when measuring on episode level than duration level, meaning that there were many short false positives. Observing performance through these six values enables a more complete characterization of the prediction performance of a certain

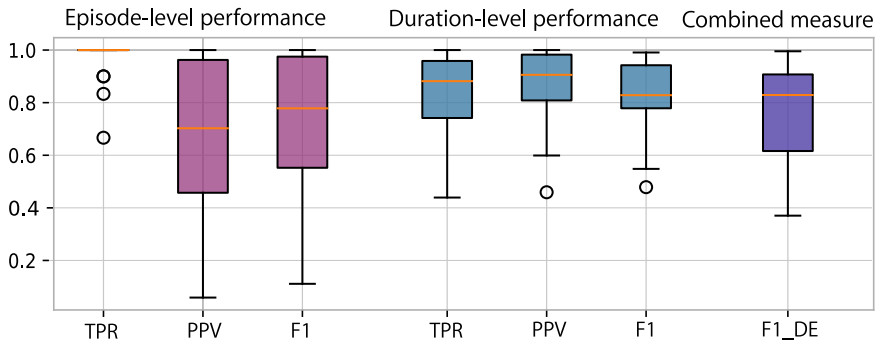

Figure 2.7: Epilepsy detection performance measured through seven measures; both on episode and duration level. For each, sensitivity (TPR), precision (PPV) and F1 score are measured. Results show average performance for all 24 subjects for 'Fact1' data subset.

algorithm. It also enables a more nuanced comparison between different methodological steps or parameter values used, as will be shown below.

2.5.2 Generalized vs. personalized models

Here we test the performance difference when training both personalized and generalized models for each subject. We used a balanced data subset ('Fact1'). In Fig. 2.10, the average for all subjects is shown, making clear the lower performance of generalized models. Inspecting the performance of the generalized model per subject reveals a clear distinction between patients on which generalized models perform very well and those for whom it performed poorly, either due to many false positives, or almost no detected seizures. How to create generalized models and whether there should be subtypes of generalized models for different patient groups remains a question for the future. For the remainder of this work we will focus on personalized models.

2.5.3 Data preparation

In Fig. 2.8, epilepsy detection performance is shown for five different data subsets used to train and test. The first two approaches, 'Fact1', and 'Fact10', contain a subset of the original CHB-MIT dataset in two different ratios of seizure and non-seizure data. 'Fact1' is a balanced data subset that has the same amount of seizure and non-seizure data, where all available seizure data is used, along with a randomly selected equal amount of non-seizure data. The 'Fact10' subset is constructed similarly, with the difference that the amount of randomly selected non-seizure data is 10x more than seizure data. The data is divided into files equal to the number of seizures with one seizure per file. Each file is arranged such that seizure data occurs in the middle of the file, with non-seizure data split on both sides. The total file length is thus dependent on the seizure length and the factor value. This organization enables easier training in the case of the leave-one-seizure-out approach, as each file is equally balanced.

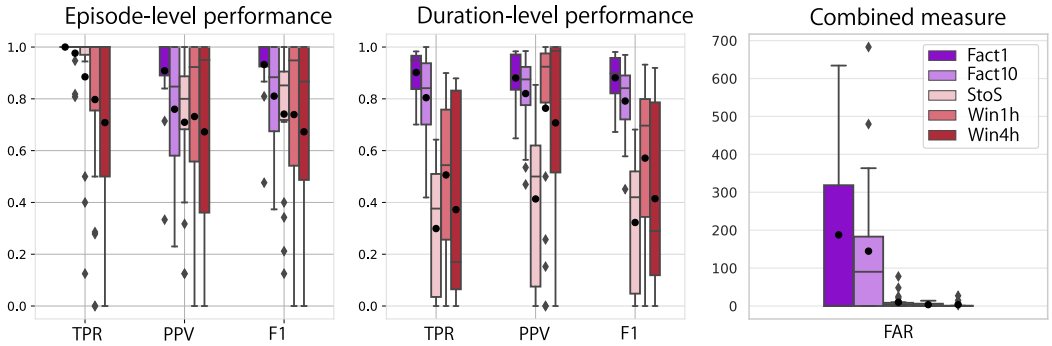

Figure 2.8: Epilepsy detection results depending on the data subset used.

The last three approaches, Seizure To Seizure ('StoS'), and 1 or 4 hour windows ('Win1h'/'Win4h', 'WinXh' together), contain all data samples from the CHB-MIT database but are rearranged into files containing different amounts of data. The 'StoS' approach consists of files that start with the non-seizure data after the previous seizure and end when the next seizure ends. In this way, every file contains exactly one seizure, but the entire length of the file is not fixed. The last two approaches, 'Win1h' and 'Win4h', as the names imply, divide the dataset into files of 1-hour or 4-hour duration. In this way, some of the files may contain zero to possibly multiple seizures. In all three cases, we trained using time-series-cross-validation. We specified that the first file must contain a certain amount of data (five hours) and at least one seizure, and as such it is slightly different than other consequent files.

We analyze the impact of data preparation by considering three metrics: false alarm rate (FAR), sensitivity (TPR), and precision (PPV). First, false alarm rate, defined here as the number of FPs per day, as shown in Fig. 2.8, is significantly higher for data subsets (Fact1(0)) than for the whole dataset training (StoS or Win1h/4h). This can be traced to two reasons. The first is that Fact1(0) is trained on much less non-seizure data, potentially not enough to model properly the non-seizure patterns, resulting in more non-seizure samples being falsely classified. The second is that because testing is done only on a subset of data, FPs must be linearly scaled to estimate the false alarm rate per day, potentially leading to very high numbers. For these reasons, data subsets should not be used for estimating the false alarm rate of an epilepsy detection algorithm.

Next, we consider sensitivity and precision. For Fact1(0), seizure episode detection is easier, visible from episode-level sensitivity that is 100%. Detecting all seizure episodes perfectly is much harder when the whole dataset is used (StoS or WinXh) visible by episode sensitivity values ranging between 80 and 95%. Precision presents a more complex picture, dropping sharply for StoS before recovering for WinXh. This is because StoS retraining and testing only occurs after every new seizure where in the meanwhile could have been hours of non-seizure data. The WinXh strategies retrain much more often, thus making it easier to learn non-seizure patterns and lower false predictions. To conclude, using all data significantly reduces false positives but also results in lower, but more realistic sensitivity and precision values.

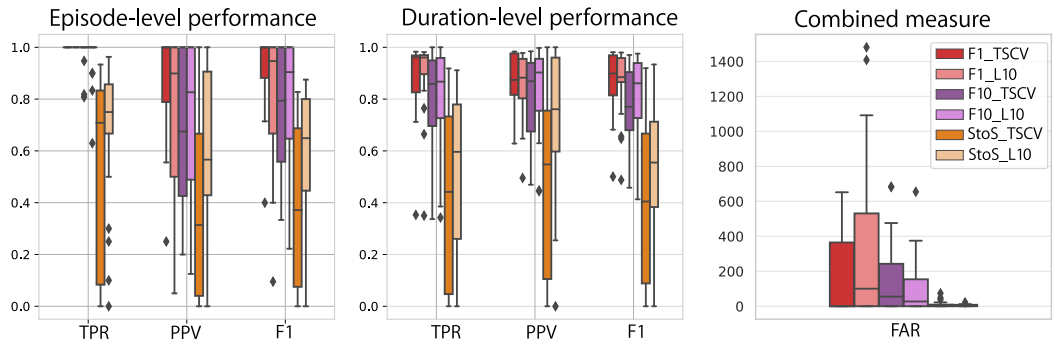

Figure 2.9: Performance results when using leave-one-out vs. time-series cross-validation. Results are shown for three data subsets (F1, F10 and StoS).

2.5.4 Managing temporal data dependencies

To show the influence of cross-validation choices on performance results, we trained and tested three data subsets with different data imbalance ratios ('Fact1', 'Fact10', 'StoS'), both using leave-one-seizure-out (L1O), and time-series cross-validation (TSCV) approaches. Results are shown in Fig. 2.9. The superior performance of the L1O approach is evident in almost all aspects (sensitivity, precision, and numFP). This is reasonable as more data was used to train with L1O than with TSCV approach. The difference in performance ranged from 3 to 7% for F1 score in episode detection. This is not a recommendation to use L1O; in contrast, it demonstrates that training on future data leads to overestimated performance and should be avoided.

2.5.5 Data segmentation

In Fig. 2.11, epilepsy detection performance is shown when the window step varied between 0.5 to 4s, with a window size of 4s. The clearest and expected pattern is that by increasing the step size, the number of false positives is reduced significantly, but what is interesting is that the proportion of FP increases, thus reducing the precision. More precisely, it increases first and then drops for episodes, leading to the conclusion that too big steps are risky. Increasing window step size also reduces sensitivity, more noticeably for the duration- as opposed to episode-level performance. This may be due to the fact that with big step sizes, shorter seizures can be potentially missed. These results demonstrate the complexity of the window step size parameter, that it is beneficial to experiment before choosing one value, and that it has to be necessarily reported to make results comparable and reproducible.

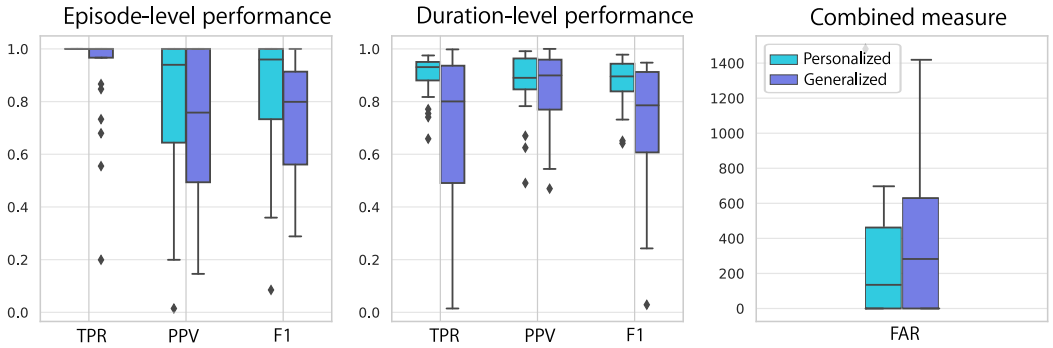

Figure 2.10: Comparison of average performance (for all subjects) of personalized and generalized models.

2.6 Discussion

2.6.1 Data aspects

As seen from results in Sec. 2.5.3, using all data significantly reduces false positives and also results in lower, more realistic sensitivity values. Thus, if computational and memory resources are sufficient, the models should be trained using all available data. If this is not possible, then the subset of data should contain significantly more non-seizure data. Even so, data subsets will result in an unrealistic false alarm rate. When using the whole dataset, the most appropriate seems to be to use fixed-size time frames in which models are retrained/updated regularly. The size of this time frame should also be tested and reported. General advice would be to use data subsets for initial experimentation and building an understanding of the algorithm and its parameters, but all the available data should be used for reporting the final performance. It is also useful to take into account and characterize the class imbalance.

When talking about the temporal aspect of the data, several things should be taken into account. We advise not to shuffle data samples before training and testing but rather to use temporal information and knowledge on class distribution to post-process predicted labels, which can increase performance, but must also be clearly reported.

Finally, it is critical to decide whether to use only the data from the same subject to create personalized and, as shown, more precise models, or to use all available data from other subjects to create generalized models. Generalized models can have lower individual performance but can be used for new subjects. This topic represents a full research topic on its own. For example, future research should investigate whether it is possible to create models that can use generalized models as a starting point from which they can be personalized.Would these models lead to better overall performance in comparison to personalized models? Would less personal data be needed to personalize models if generalized modes are used as starting point? Similarly, the unavoidable question is, can we somehow profit from both generalized and personalized models? Can we combine them in some beneficial way?

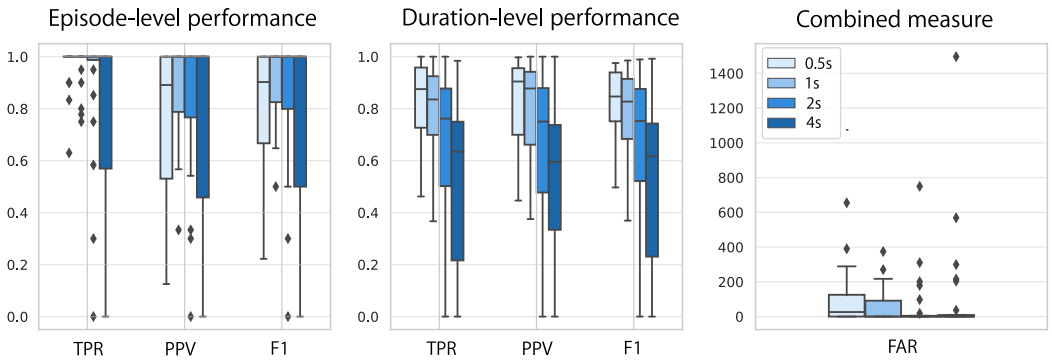

Figure 2.11: Performance with respect to different window step sizes. Window step sizes of 0.5, 1, 2 and 4s were used, with window size of 4s.

2.6.2 Training aspects

When talking about the choice of cross-validation as shown in Sec 2.5.4, the leave-one-out approach leads to higher performance than if data is trained in temporal order using time-series cross-validation. However, the L1O approach is not realistic for training data in real-time, while TSCV is intended for such scenarios. Training models online as data is being acquired is one of the necessary next steps for ML models on IoT devices, and thus TSCV will have to be become the standard method.

Data partitioning has two parameters that can also play a significant role in performance, namely the window size used to extract features and the window step size. Their optimal choice can depend on each use case, the features extracted and their properties and complexity, latency requirements, and available computational resources. Here we showed how window step size can influence performance, with different patterns for false alarms, sensitivity, and precision, and how it has different impacts on duration- or episode-level classification. Results show the complexity of the window step size parameter, indicating that it is beneficial to test it before choosing one value, and that it must be reported to make results comparable with the literature. One research avenue we have not considered here but which can be potentially very beneficial is optimizing the window size parameter for each feature individually.

2.6.3 Performance estimation aspects

Here we proposed to use two performance metrics, one at the duration-level and one at the episode-level. Each of them has certain advantages, and thus their values should be interpreted carefully. Nevertheless, together they provide a full picture of the detection characteristics of the algorithm analyzed.

For example, EPOCH, a duration-based metric, cares about the duration of the events and thus weighs long events more importantly. This means that if a signal contains one very long seizure event and some shorter ones, the accuracy with which the long event is detected will

dominate the overall scoring. In epilepsy detection, as in many applications, event duration can vary dramatically; thus, this has to be kept in mind. For this reason, OVLP, an episode-level performance metric, is much easier to interpret. However, such episode-level metric is more permissive and tends to produce much higher sensitivities. It can also be implemented so that if an event is detected in close proximity to the reference annotation, it is considered correctly detected, which can further increase the performance values.

Nowadays, in the literature, duration-level-based performance is still the most popular, but there are trends of moving towards more event/episode-based performance measures [244]. As of yet, there is no standardization. Until then, the performance metrics used, as well as the post-processing that has been utilized to smooth the labels, must be clearly described.

Similarly, the method of attaining the overall performance measure from the individual CV folds must be documented. We recommend that overall performance be calculated via temporally appending all fold predictions in time, rather than as the average of all fold performances. This is due to potential specific patterns in the distribution of false-positives (such as all in one file) that can lead to overestimated performance when reporting the average.

2.7 Conclusion

In this chapter, we give an overview of the continuous EEG and iEEG epilepsy datasets available to researchers, some of which are short-term and some are true long-term recordings. A thorough characterization of datasets was done, taking into account multiple parameters of the recordings (number of subjects, channels, sampling frequency, biosignals recorded, etc.) as well as the extent of annotations (seizure type, onset and stop, patient information, etc.). We defined several critical characteristics that make them suitable for certain model types or methodologies. We also commented on how this can influence further progress in the development of wearable devices for the monitoring of epilepsy.

Furthermore, we characterized the influence of a broad range of methodological choices important for epilepsy detection systems. When choosing a subset of the dataset for training, performance can be highly overestimated compared to training on the whole long-term dataset. Thus, for real-life performance estimation, using entire long-term data is necessary. Similarly, using the leave-one-seizure-out cross-validation approach can improve detection performance but it is not realistic for online data training as it uses future data. Thus, we recommend using a time-series cross-validation approach instead, with macro-averaging rather than micro-averaging.

Training on a generalized level can be challenging due to its subject-specific nature, leaving personalized models outperforming generalized ones. But ultimately, generalized models will be a necessary part of future wearable devices. Furthermore, performance metrics must reflect users' needs and be sufficiently sensitive to guide algorithm development. For this reason, we encourage the usage of both episode-based and duration-based performance metrics, which

can together give a more nuanced picture of algorithm performance. Finally, whatever choices are made, to further increase the comparability and reproducibility of results, it is essential that all choices and parameters are well reported.

Limitations

All the analysis in this chapter was performed using random forest as a model. We believe that similar trends would be present for different ML models as well, but this could be thoroughly examined in practice, too. Furthermore, we tested numerous methodological choices, but there are additional ones that can be tested, such as the influence of data under and oversampling as their combination.

Moreover, epilepsy-specific, researchers often exclude preictal or postictal data from training or even testing sets. As these recording sections often exhibit patterns similar to those of the seizures themselves, this could represent the most challenging data to train and test. Thus, it would be interesting to quantify how much this exclusion (only from the train, only from the test, or from both sets) can influence detection performance.

3 Encoding spatio-temporal data

Spatio-temporal (ST) data is characterized by sparsely distributed data observations in space and time. It can be divided into different categories [307] such as event data (discrete events occurring at point locations and times), trajectory data, raster data (data recorded at fixed locations in space and at fixed time points), or video data. Video can even be considered a special type of raster data. But perfect examples of raster data are various recording modalities used in neuroscience, such as functional Magnetic Resonance Imaging (fMRI), electroencephalography (EEG), Magnetoencephalography (MEG), and functional Near Infrared Spectroscopy (fNIRS). These technologies capture data with different spatial resolutions in/over the brain and over a certain period of time, but the resolution can significantly differ between technologies. For example, fMRI measures neural activity from millions of locations, while EEG only records from tens of locations. On the other hand, fMRI typically measures activity every two seconds, while the temporal resolution of EEG data is typically one millisecond.

In recent decades, our understanding of the brain-behavior relationship has improved dramatically, mainly due to brain imaging techniques and recordings with good spatial and temporal resolution [308], [309]. For example, without imaging methods, localization of epilepsy onset regions would not be possible, and only with such knowledge and technologies can clinicians plan and execute surgeries.

However, analyzing such multidimensional data is challenging because it requires modeling temporal and spatial correlations to determine the most discriminative features for a specific application. Classical machine learning approaches often perform poorly when applied to spatio-temporal datasets [307]. Some of the reasons are that many models are not made to model continuous data, and also that they are limited by the common assumption that data samples are independently generated, which is not correct for ST data. Moreover, data correlations existing in ST recordings are hard to capture by traditional methods. Although progress has been made in the last decades in capturing information from ST data, it is largely based on feature engineering. In other words, conventional machine learning and data mining techniques for ST data are limited in their ability to process natural ST data in their raw form.

Researchers put a lot of focus on deep learning methods due to their ability to automatically learn hierarchical feature representations from raw data. Convolutional neural networks were designed to process image data and are now also widely used in mining ST data. GraphCNN is studied to generalize CNN to graph-structured data [310]. On the other side, recurrent neural networks have been designed to recognize the sequential characteristics and predict the next data point, thus finding lots of applications for speech recognition and natural language processing. For this reason, they are interesting for temporal data. The limitation of RNNs is that they have quite short-term memory due to the issue of vanishing gradients. Thus, Short-Term Memory (LSTM) networks have been designed as an extension for RNNs, and are capable of learning long-term dependencies of the input data. Hybrid models combining both CNN and RNNs have been proposed. For example, CNN and RNN can be stacked to learn the spatial features first and then capture the temporal correlations among the historical ST data [311].

However, to the best of our knowledge, encoding spatio-temporal data (such as EEG data) in the scope of HD computing has not yet been properly discussed. In most of the existing literature utilizing EEG or EMG data, only raw data or *local-binary-patterns* (LBPs), [312], have been used as features encoded to vectors. Yet, similarly to standard ML approaches, the possibility of adding more features can significantly improve the power of the models, as will be also shown in this chapter. Thus, here, we discuss possibilities for encoding information about channels, time, and features to capture all relevant correlations and aspects of the data.

3.1 Dealing with spatio-temporal data

3.1.1 Methods

When dealing with long recording of time series data, it is essential to adopt an assessment methodology that represents real-life usage to avoid achieving unrealistic performance metrics. Therefore, the use of only a subset of a dataset to assess the performance of the proposed ML algorithms should be avoided, whenever possible. Even if models are trained using a small subset of the dataset, the evaluation should be done on the available data in general. For example, as demonstrated in Chapter 2, the performance of some works such as [36], [241], [313] tends to be overestimated and impractical when considering the entire data set, as only a small number of data points are used to assess the proposed techniques. This happens as the probability of drawing samples of data presenting artifacts and other common acquisition problems randomly is very low. Thus, if possible, the whole dataset containing the original data distribution should be used.

Time-series cross validation

Furthermore, due to the existing time relation between recordings that can play a significant role in pattern detection, we advocate for the usage of a cross-validation (CV) method that

takes into account the chronological relation between data points. This approach preserves the temporal dependency among data and thus represents the assessment of a real-life application more appropriately.

For instance, in the leave-one-out CV (LOOCV) or for personalized seizure detection leave-one-seizure-out method future data is used to evaluate the current test set, leading to often over-estimated results. To overcome this problem, we propose the Time-Series-Cross-Validation (TSCV) approach [301], which is illustrated in Figure 2.5 and was introduced in Sec. 2.3.3. We sometimes call this approach also Rolling-Base (RB) approach. In this CV scheme, we also set a rule that a minimum amount of data is used to train the very first model, which must include at least one seizure. More specifically, we use at least 5 hours of initial data to train the first model. The remaining data is used for testing and divided into several subsets of a fixed length of files. We tested different lengths of files and concluded that a minimum of one-hour long files presents a good optimum. It minimizes potential correlation with neighboring training data but also allows for a number of cross-validations for large datasets. Each of these subsets is tested once, and at each iteration, the previously tested data is added to the training subset. The final performance is then obtained by concatenating the results of each test subset.

Label post-processing

Due to available information on the time order of samples, a post-processing step that takes into account statistical distributions of labels is possible. This process smoothes predicted labels to correspond to more realistic label distribution. For example, for epilepsy detection, it is not realistic for seizures to last only a few samples, and two seizures closer than a few seconds are usually considered the same seizure.

In the literature, there have been two label post-processing methods used on epilepsy datasets:

- **Moving average smoothing**: This method takes into account the last W predicted labels and 'smoothes' them by classifying them as seizure only if a certain percentage of labels was also 1. More specifically, the classification likelihood of seizure (CL) is calculated as expressed by Eq. 3.1, and if the value is higher than, e.g., 0.5 the label of sample i is 1. This approach has been regularly used in the literature [204], [240], [314]. Some works used fixed thresholds for all subjects, whereas some even personalized thresholds for each subject [314].

$$CL[i] = \frac{\sum_{j=i-W}^{i} Labels[j]}{W} \tag{3.1}$$

- **Bayesian smoothing**: In this method, instead of using only previous labels, label probabilities are used. The cumulative probability that it is seizure $ProbS$ or non-seizure $ProbNS$ is calculated for a window of the last W samples. Classification likelihood (CL) for a given window is defined by Eq. 3.2 and compared with the threshold. If the probability exceeds the threshold, the label of 1 is given to the current postprocessed

sample. This approach has been proposed in [280] for epilepsy prediction processing and is thus tested in this thesis.

$$CL[i] = \log_2 \frac{\sum_{j=i-W}^{i} ProbS[j]^2}{\sum_{j=i-W}^{i} ProbNS[j]^2} \tag{3.2}$$

Further, sometimes a second step of post-processing is practical. Namely, if two seizures are very close (e.g., 30 s) seizures could be merged (i.e., all predictions between them are flipped from 0 to 1). Due to the small window length and even smaller moving step, granularity is much smaller than the dynamics of the seizures, so these steps lead to more realistic and robust seizure predictions.

Performance evaluation

In order to capture as much information as possible about the model performance, we recommend measuring performance on two levels: 1) episode level and 2) seizure duration level. This is in detail explained in Sec. 2.3.5. Metrics on these two levels give us better insight into the operation of the proposed systems. These types of metrics are also advocated by the clinical community [90]. Furthermore, the performance measure often depends on the intended application and plays a big role in the acceptance of the proposed technology.

For both levels, several standard metrics can be assesed; e.g., sensitivity (true positive rate or TPR, calculated as $TP/(TP+FN)$), precision (positive predictive value or PPV, calculated as $TP/(TP+FP)$) and F1score ($2*TPR*PPV/(TPR+PPV)$).

3.1.2 Adapting HD computing for spatio-temporal data

Spatio-temporal data is highly complex and usually contains valuable information hidden in temporal and spatial relationships. Two approaches to capture this information are: 1) through feature engineering and 2) through ML model structure that supports capturing temporal, spatial, or even both information.

Encoding time through features

In the first subchapter of this chapter, we compare different existing literature features for epilepsy used with classical ML, but now employed with the HD computing paradigm. Many of these features capture time component through frequency analysis, entropy, or various decomposition techniques. As so far, in the literature, only raw data or data trends (LBPs) have been encoded to HD vectors for epilepsy detection; we want to analyze if there is an advantage to encoding expert features. Furthermore, since in the literature methodologies are not consistent, here we perform analysis in a systematic way keeping all parts of the workflow the same except features we are testing.

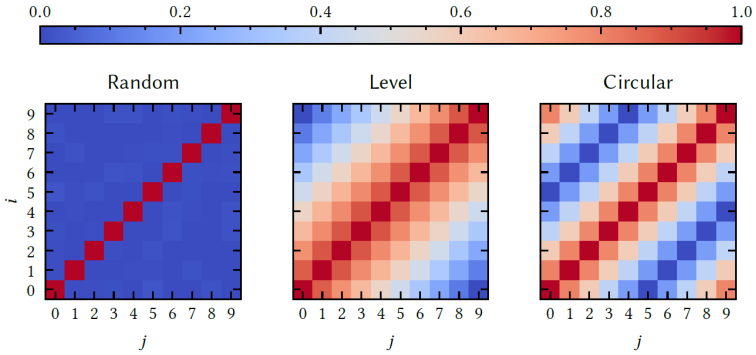

Figure 3.1: Three possible types of initialization of base hypervectors. Illustration is taken from [315].

Then in the next subchapter, we focus on designing a new interpretable feature that considers the frequency, time, and amplitude of the signal. Observing how neurologists inspect signals and label data inspired us to design a feature that mimics their pattern-detecting strategies. We demonstrate that indeed by using a highly specific feature that captures time and amplitude information, it is possible to outperform models with many classical literature features.

Encoding spatial information

In the last chapter, we explore how spatial information can be included as part of the HD workflow. More specifically, we propose and test different ways to encode spatial information to vectors. Epilepsy detection performance is compared with approaches taking spatial information into account and approaches ignoring spatial information. Memory requirements and computational complexity of each encoding approach are also analyzed, as this plays an important role in wearable implementations.

A similar approach for embedding time information during the encoding process can be analyzed in the future. In fact, some researchers utilized vector permutation [204] for including time information while encoding raw data. But overall, more systematic analysis is also necessary for time encoding, similar to what we perform here for spacial information.

In the end, important to note is the initialization process of *base* (or also called *basis, seed* or *atomic*) vectors. As explained in Sec. 1.3.2, these base vectors represent 'units of information' or different information types that we want to encode (e.g., features, values, channels, etc.). During encoding one data sample, these base vectors are combined using several arithmetic operations also presented in Sec. 1.3.2. Since the nature of these basic units of information is different, so should be the base hypervector sets used to encode them. This has been nicely illustrated and discussed in a recent paper [315]. Namely, as already briefly mentioned, vectors that initialize, e.g., all possible values of features, can be initialized randomly, but also in a way that maps distances between values they represent. Thus in [315], authors present three ways of initializing vectors: random, level, and circular, as illustrated in Fig. 3.1. More specifically:

- **Random initialization**: Random initialization means that each vector is initialized uniformly and randomly from the hyperspace. This results in vectors that are with high probability orthogonal and, as such, good for representing information sets that do not have any specific relation between them, i.e., categorical/symbolic data such as letter symbols or, in our case, features.
- **Level initalization**: It is used to encode linearly correlated information (e.g., distance, time, etc.), in general, feature values. Unlike random-hypervectors, these vectors are usually generated using an interpolation of vectors representing the smallest and biggest value, which are randomly sampled. For binary vectors, interpolation is done using bit flipping of a portion of vector bits that correspond to the distance between the minimal and current vector.
- **Circular initialization**: Less commonly used, but still possible, are initializations where information relation is specific: e.g., circular. Examples of such information types are angles, time of the day, seasons of the year, etc. In this case, vectors are initialized and created to map distances in both directions.

A detailed description of the process of creating these types of initializations and an in-depth study of their characteristics is available in [316]. In this thesis, we use random initializations for feature vectors and usually level initialization for feature values. When discussing spatial representation, research should be done on the optimal initialization of channel base vectors. In cases when channels location are not really known (such as in iEEG datasets), the only option is random initialization, but in cases with known localization on the head, more specific EEG-based initialization could be tested, which will map 2D/3D relations between all channels. Until then, we use random initialization for channel vectors as well.

3.2 Encoding features to HD vectors

3.2.1 Motivation

The first work that applied HD computing to epileptic seizure detection [206] relied on mapping local binary patterns (LBPs) to HD vectors. LBPs map a sequence of data samples onto a small binary array, relying solely on whether the amplitude of the data increases or decreases. The authors encode LBPs into HD vectors and test the approach on SWEC-ETHZ iEEG database. They focused on testing one-shot learning, or learning from as few seizure instances as possible. They showed that for most patients, the algorithm learned from one to two seizures and achieved perfect specificity and sensitivity.

In another recent paper, HD computing was applied on EEG data, which is more viable for continuous long-term monitoring [204]. The authors also compared HD computing with different standard state-of-the-art ML approaches (KNN, SVM, regression, random forests, and CNN). For KNN, SVM, regression, and random forests, they used 54 different features from [241] and [36], while for the HD computing approach, raw amplitude values were encoded into HD vectors. More precisely, they normalized data and mapped amplitudes of all samples to the corresponding HD vector. The authors used the CHB-MIT and reported that the HD approach surpassed the performance of all other approaches.

Both aforementioned papers present promising results for the application of HD computing for epileptic seizure detection, but results cannot be directly compared. Namely, data preparation (filtering, train/test split), segmentation (size and step of discretized windows), mapping to HD vectors (how channel and time information is encoded) as well as used performance metrics are different in both papers. Moreover, when comparing with different ML models, authors of [204] used different features making comparison even harder.

Thus, the main motivation of this work is to perform a systematic assessment of the HD computing framework for the detection of epileptic seizures. In particular, we compare different feature encoding strategies on HD vectors by evaluating them in a comparable way using a common workflow (i.e., with the same pre-processing setup and identical performance measures). We perform the analysis on two different datasets to assess the generalizability of our conclusions.

Thus, we contribute to the state of the art in the following manner:

- We investigate and systematically compare different several literature features and methods of encoding them to HD vectors in terms of prediction accuracy (sensitivity, recall, and F1) for both seizure episodes and duration, as well as in terms of computational complexity and memory requirements.
- We implement new features that are often used in machine learning techniques for the detection of epileptic seizures but that have not yet been implemented in HD computing.

- We utilize two publicly available datasets to improve the generalizability of our methodology.
- Among our results, we demonstrate that post-processing significantly reduces the differences between approaches. However, our results show that there are strong trade-offs between approaches in terms of memory footprint and computational complexity.

3.2.2 Data encoding to HD vectors

In this section, we detail the different signal features that may be exploited for epileptic seizure detection and how they are encoded into HD vectors.

Encoding methodology

First, HD vectors that represent different features HDV_{Feat}, feature values HDV_{Val}, and channels HDV_{Ch} are created at the beginning of training in the form of a memory map, enabling simple one-to-one mapping between values and corresponding HD vectors (see Fig. 3.2). This means that if we have 10 features, 10 HD vectors are initialized, one representing each feature. Similarly, for channel vectors, one vector is initialized for each channel. For feature values, values are normalized and discretized to a certain number of levels, so that one vector is assigned to each normalized feature value.

In this work, we use an approach where all HDV_{Feat} and HDV_{Ch} HD vectors are independently and randomly generated at the beginning of the training. On the other hand, the vectors HDV_{Val} were initialized in a way in which the vector was initially randomized, but then every subsequent vector representing the next possible value was created from the previous by permuting consecutive blocks of d bits. The number of bits d depends on the number of possible values (and corresponding HDV_{Val} vectors) that are needed. This approach ensures that vectors representing numbers that have closer values are also more similar.

Next, feature encoding is performed by first calculating the feature(s) of interest for a given discrete window of size W_{len}. Then, for each feature value, an HD feature vector HDV_{Feat} representing that feature, an HD value vector HDV_{Val} representing the value itself, and the corresponding channel vector HDV_{Ch} are selected.

Since vectors are binary vectors, we bind vectors by using bit-wise XORs between each feature vector HDV_{Feat} and the corresponding value of that feature vector HDV_{Val} or channel vector HDV_{Ch}. All bound vectors are then bundled up (bit-wise summed (SUM) and rounded), resulting in an HD vector representing each discrete window. Depending on the feature approach used, binding and bundling of data is slightly different, as described in the next Sec. 3.2.2, as well as illustrated in Fig. 3.2 (sub-windows A, B, C and D).

Figure 3.2: Schematic of the HD computation workflow. The only part that is different for different feature sets that are used is part of encoding data/features to vectors (marked with *). For encoding raw data or data trends A) approach is used. For encoding complex features like FFT B) method is used. Finally, for individual features, method C) is used, and for multiple features, method D).

Features to encode

Local Binary Patterns

The *LBP* approach maps a sequence of raw data into arrays of 0's and 1's. Namely, if the signal between two samples is increasing, the bit corresponding to that position has a value of 1, otherwise 0. This procedure maps the changes in the signal trend but not the values of the signal itself. The distribution of LBP codes within ictal and interictal states are different and pose an interesting feature to use: during the interictal states, LBP codes are almost evenly distributed over all possible codes, while in ictal windows certain codes are more dominant [206]. Each data window is represented by encoding LBP features into an HD vector. First, we calculate LBP binary patterns for each sample and map these binary vectors to the corresponding HDV_{Val} HD vectors. We bind (XOR) HDV_{Val} vectors with the corresponding HDV_{Ch} channel vectors, yielding HDV_{ValCh} vectors. HDV_{ValCh} vectors are then bundled (summed and rounded) all together. This is repeated for all samples in each W_{len} of data and bundled to represent that window as illustrated in Fig. 3.2(A). The length of the LBP patterns can be chosen arbitrarily, but here we use 6-bit patterns, as proposed in [206]. It was demonstrated in [239] that this value leads to the best performance, and this also enhances comparability between papers.

Raw Signal Amplitude

The raw signal amplitude (*RawAmpl*) approach is similar to the one of LBP because it does not rely on extracting any complex features from the data, but solely encodes normalized raw signal amplitudes into HD vectors. Namely, here we use a simpler version of the encoding from [204] to make it more comparable to other approaches. First, we normalize the raw amplitude of every sample. Then, we map the values to the corresponding HDV_{Val} vectors and bind (XOR) together with HDV_{Ch} HD channel vectors, yielding HDV_{ValCh} vectors. This is repeated for each sample in a window. Finally, the HD vector representing the current window is computed by bundling (summing and rounding) all bound HDV_{ValCh} vectors across time samples of that window.

Frequency Spectrum

Since frequency spectrum components are common features used for epileptic seizure detection, we tested whether encoding the FFT spectrum into HD vectors can be useful for classification. The *FFT* approach we use is similar to the one in [187], i.e., applied on a voice recognition task. Here, as illustrated in Fig. 3.2(B) we calculate the FFT spectrum for each W_{len}. Then we bind (XOR) together normalized FFT values HDV_{Val} with corresponding frequency HD vectors HDV_{Freq}, yielding $HDV_{ValFreq}$ vectors. This is repeated so that vectors for all channels are bundled (summed and rounded) together to get an HD vector representing that data window.

Single Features

Inspired by the simplicity of features from previous approaches, we also evaluate if single features calculated on W_{len}, and encoded to HD vectors can achieve similar performance as *RawAmpl* and *LBP*. We test three commonly used features in state-of-the-art ML algorithms for epilepsy detection:

- *Mean amplitude* is defined as the mean amplitude of the normalized signal in window W_{len}.
- *Entropy* is defined as the normalized spectral entropy value in window W_{len}, and is defined as:

$$H(x, sf) = -\sum_{f=0}^{fs/2} PSD(f) * log_2[PSD(f)] \tag{3.3}$$

  where PSD represents the normalized Power Spectral Density of the data, and fs the sampling frequency.
- *Continuous Wavelet Transform (CWT)* is defined as the area of the frequency spectrum with the highest energy. More precisely, we calculate the CWT decomposition of signal window W_{len} with 20 levels in the range [0.25,15] Hz. The index of the frequency band with the highest energy is chosen as a feature.

For each feature, the value of the feature is assigned to the HD vector HDV_{Val} and bound (XOR) with the corresponding channel vector HDV_{Ch}. Finally, to obtain the HD vector representing a W_{len} window of data, we bundle (sum and round) HDV_{ValCh} vectors of all channels together, as shown in Fig. 3.2(C).

Combining Multiple Features

Inspired by machine learning approaches where more features lead to performance improvements [241], we combine multiple features in a single model. We test this in two steps:

- *3Feat*: utilizing the three above-mentioned features (mean amplitude value, entropy, and highest energy frequency band), and
- *45Feat*: using the 45 features proposed by a standard ML approach to seizure detection [241].

These features contain 37 different entropy features, including sample, permutation, Renyi, Shannon, and Tsallis entropies, as well as 8 features from the frequency domain. For each signal window, we compute the power spectral density and extract the relative power in the five common brain wave frequency bands; delta: [0.5-4] Hz, theta: [4-8] Hz, alpha: [8-12] Hz, beta: [12-30] Hz, gamma: [30-45] Hz, and a low frequency component ([0-0.5] Hz). These features are commonly considered medically relevant for the detection of seizures [317].

For each feature, its value HDV_{Val} and its index vector HDV_{Feat} are bound (XOR), to get

$HDV_{ValFeat}$ vectors. Finally, to get a final HD vector representing each W_{len} , we bundle (sum and round) $HDV_{ValFeat}$ vectors of all features and channels, as shown in Fig. 3.2(D).

3.2.3 Experimental Setup

Databases
We assess the studied approaches on two publicly available databases: CHB-MIT for EEG data [242], [243] and SWEC-ETHZ for iEEG data [206], as described in Sec 2.1.2. We are interested to see if the trends for different performance measures will be the same for both databases.

Dataset Preparation
Several steps are required to make the databases comparable and applicable for epileptic seizure classification. First, both databases are (re)sampled to 256Hz. Further, we create balanced files for each seizure, where, in each file, there is an equal amount of interictal (non-seizure) and ictal (seizure) data. More specifically, from the original database files, for each file containing ictal data, we keep all ictal data. Before the onset of the seizure, we keep the same length of randomly selected interictal data. We exclude interictal data that is within 1 minute of seizure onset and up to 15 minutes after a seizure, as this data might contain ictal patterns. Balancing the data allows us to focus on assessing the separability between classes and prevents issues related to the prior distributions among classes.

Validation
Given the subject-specific nature of seizure dynamics, we evaluate the performance on a personalized level. As already mentioned, the data for each subject is divided into files, where each file contains one seizure and the same amount of non-seizure samples. This enables us to use a leave-one-seizure-out approach, where the HD model is trained on all but one seizure/file, while at the same time ensuring a balanced dataset for both training and testing. For example, for a subject with N_{seiz} files (each containing one seizure), we perform N_{seiz} leave-one-out cross-validations and measure the final performance for that subject as the average of all cross-validation iterations.

Computational and Memory Complexity Evaluation
In order to characterize all the relevant aspects that could determine the feasibility of future implementations on embedded systems, we also study the memory requirements and computational complexity of the proposed feature approaches.

Memory requirements are assessed by calculating the memory needed to store all prototype vectors for each particular approach (e.g., memory for HDV_{Val} vectors, HDV_{Feat} vectors, HDV_{Ch} vectors, etc.). We express it relative to the dimensionality of the HD vectors D, i.e., actual memory size is calculated by multiplying by D (number of values in each vector) and its type size (bit/integer/float). On the other hand, the computational complexity is estimated by calculating the number of SUM and XOR operations and measuring the time required for

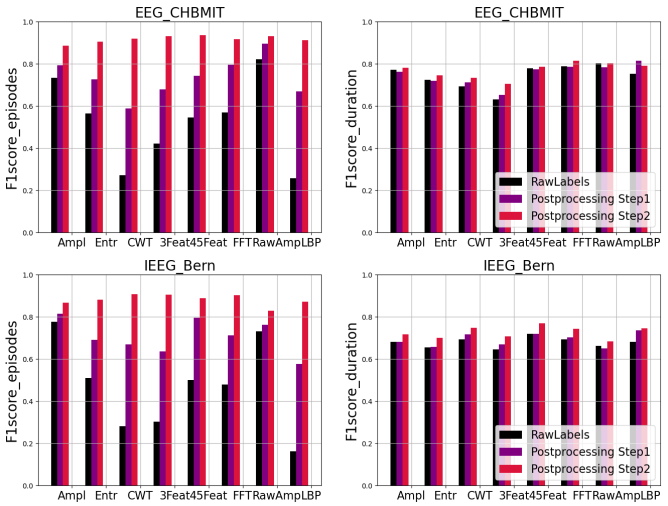

Figure 3.3: F1 score performances for different feature approaches, both for episodes and duration level and two databases.

feature calculation and encoding to an HD vector. It is measured per one discrete data window W_{len}, as all analysis and predictions are window-based. Time is assessed as the average time over 1000 discretized windows for both datasets and all feature approaches.

3.2.4 Results

Prediction performance

Fig. 3.3 shows F1 score of epileptic seizure detection before and after two steps of post-processing for both databases. Performing moving average with voting in the first step improves performance at the episode level (up to 41.2% for EEG and 41.5% for iEEG). However, adding another step to merge close seizures improves the performance even more (up to 65.5% for EEG and 70.9% for iEEG). On the other hand, post-processing steps provide a minor improvement on the seizure duration level (up to 4.0% for EEG and 6.4% for iEEG after both steps). Papers in the literature use different post-processing approaches (often without clearly mentioned parameters); thus, a careful comparison among different works is difficult.

We compare feature approaches primarily before label post-processing, separating it from the influence of post-processing. At the episode level, the best performance is achieved by approaches including amplitude information (*RawAmpl* and *Ampl*), while *Entropy* and *CWT* perform worse than *Ampl*. Interestingly, combining the three features does not lead to a performance improvement over just using individual features themselves. Further, using 45 features *45Feat* rather than three *3Feat* leads to an improved performance, but not better than *Ampl* alone. *FFT* performs on the level of 45Feat, while *LBP* performance is much lower. The range of F1 score performance before post-processing for different feature approaches is [25.8%-82.2%] for EEG, and [16.3%-77.7%] for iEEG. This is due to the large number of false positives that exist before post-processing. After post-processing, differences between approaches are significantly reduced, i.e., with range [88.6%-93.7%] for EEG and [82.8%-90.7%] for iEEG.

Table 3.1: Memory and computational requirements of different sets of features.

Features	Memory (fact. of D)	Number of operations	CHB-MIT Comput. time [ms]	CHB-MIT Ratio Feat vs. HD vec	SWEC-ETHZ Comput. time [ms]	SWEC-ETHZ Ratio Feat vs. HD vec
Ampl	36	32	0.00093	1.809	0.001	1.657
Entropy	36	32	0.0061	15.618	0.0064	14.690
CWT	36	32	0.3343	690.328	0.3277	617.180
3Feat	23	112	0.3769	33.544	0.3555	34.788
45Feat	65	1456	0.3766	33.988	0.3530	31.525
FFT	84	2064	0.0108	0.583	0.0109	0.574
RawAmpl	36	33792	0.5788	0.787	0.5573	0.777
LBP	80	33792	0.3883	0.587	0.3787	0.583

Performance on the level of duration is, on the other hand, less variable between approaches, with the F1 score ranging [63.0%-80.4%] for EEG and [64.4%-72.0%] for iEEG, before post-processing. Moreover, improvement with post-processing is smaller, i.e., performance is [70.4%-81.4%] for EEG and [68.3%-76.9%] for iEEG. The relative performance between different feature approaches has very similar trends on both databases, which makes results more generalizable in the scope of epileptic seizure detection.

Finally, when comparing the final F1 performance after two steps of post-processing, approaches differentiate in 5.1% and 11.0% for episodes and duration of EEG database, and similarly, 7.9% and 8.6%, for iEEG database. Here, we see that after post-processing, approaches differ less on the level of episodes, even though they differ more before post-processing. This means that the best choice of feature approach to use, depends not only on the amount of post-processing but also the performance objective (i.e., only seizure episode detection or also seizure duration detection).

Computational Complexity and Memory Requirements

The complexity and memory requirements results are shown in Fig. 3.4 and Table 3.1. Memory requirements for storing HD vectors are different among approaches, and depend on the exact number of channels, features, segmentation levels for feature values, and sampling frequency.

The memory requirements are expressed in Table 3.1 as a factor of the dimensionality of HD vectors (D). *LBP*, *FFT* and *45Feat* approaches have biggest requirements; *LBP* due to the large number of possible *LBP* values, *FFT* due to high frequency resolution and *45Feat* due to the large number of features.

In terms of computational complexity, variance is very high between different approaches. The most computationally expensive is *RawAmpl*, followed by *LBP*, and then *45Feat*, *3Feat* and *CWT*, but the reasons for this complexity are different among approaches. In Table 3.1, the ratio of time required for feature calculation vs. time to perform all operations on vectors, as

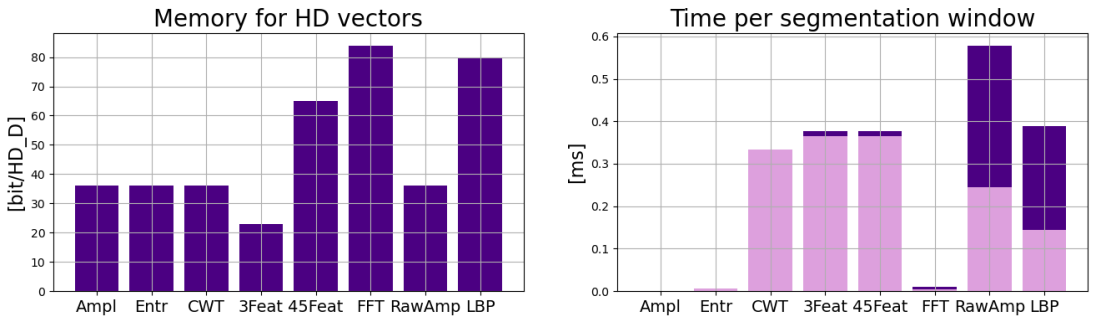

Figure 3.4: Memory and computational complexity for different feature approaches. The right graph represents the total time for processing one discrete window of data. The pink color represents the amount of time needed for calculating the feature values and the blue portion is the time required for operations on HD vectors.

well as the total time, is shown. For example, *RawAmpl* and *LBP*, even though they are simple to calculate, are sample-based. Thus, due to their high sampling frequency, the number of operations in one discretized window is very high (33792 XOR+SUM operations vs. 32 for single feature approaches). This is also visible in Fig. 3.4, as a big portion of the bars are blue, representing time needed for operations on vectors, i.e., 56% for *RawAmpl* and 63% for *LBP*. On the other hand, *CWT*, *3Feat* and *45Feat* are more computationally consuming due to the complexity of the calculated features. As shown in Fig. 3.4, a majority portion of the corresponding bars (>97%) are pink, representing feature value calculation. In contrast, *Ampl*, *Entropy*, and *FFT* are very computationally efficient, both due to a lower number of operations on vectors and a simpler feature calculation.

Overall, the ratio between the best and worst-case memory requirements are 3.8×, 1056× for number of operations, and 622× (EEG) and 557× (iEEG) for the computation time. Thus, differences between approaches are much smaller concerning memory than for computational complexity.

It is important to note that this analysis only compared aspects of memory and computation related to HD computing and HD vectors. In reality, feature extraction can also require significant amounts of memory footprint and be really time consuming, as differences can be significant between different features. Thus, for the final application, the energies of the individual features will be needed. Recent works [318], [319] even designed self-aware approaches that were able to extend battery life using a smaller or larger set of features depending on their confidence and energy efficiency. However, since here we focus only on comparing HD approaches, we did not analyze feature extraction steps before the HD part, and this is something to do before implementation on a wearable device.

Finally, our results show that approaches with high performance (such as *RawAmpl* or *45Feat*) are not ideal for wearable applications due to high memory or computational requirements. At the same time, similar but simpler approaches (e.g., single feature for *Ampl*) can maintain

high precision and require significantly less memory and computational resources. Thus, for HD computing, similar to standard ML, feature selection is an extremely important step to properly optimize the performance and hardware implementation feasibility.

3.2.5 Conclusion

In this section, we performed a systematic assessment of the HD computing framework for the detection of epileptic seizures. In particular, we compared different feature encoding strategies on HD vectors. Also, as HD computing is a novel approach and there is a lack of guidelines on strategies, parameters and calculation steps, we systematically analyzed the workflow in a practical application, with emphasis on feature encoding and multi-channel fusion. More precisely, we tested two known feature methodologies (*LBP* and *RawAmpl*), as well as several novel approaches for HD computing on epileptic seizure detection (Single Feature - *Ampl*, *Entr*, *CWT*; Multiple Feature - *3Feat* and *45Feat*, and *FFT*).

In our experiments, we compared three main aspects: 1) detection performance, 2) memory requirements and 3) computational complexity. Results show a difference in F1 score between approaches (up to 61.4% for episodes detection and up to 17.4% for seizure duration), but also that the highest performance approaches are not ideal for wearable applications due to their high memory or computational requirements. Nonetheless, after post-processing steps for smoothing prediction labels to adjust predictions to the dynamics of epileptic seizures, performance is improved and differences are significantly reduced between approaches (up to 7.9% for episodes detection and up to 11.0% for seizure duration). In terms of memory and computational requirements, differences between approaches are much smaller concerning memory (3.8×) than for computational complexity (622×). Thus, for a wearable implementation, feature selection and decisions based on several aspects are necessary. All the analysis results were performed and confirmed on two different datasets, which validates further the generalizability of our results.

Overall, these results point in two directions. Using only LBP or raw amplitude features is not necessarily ideal, and using more epilepsy discriminant features can be useful both from performance, memory, and computational aspects. But those features have to be well designed to be highly discriminative for epilepsy detection. And secondly, for time-series data, post-processing plays a big role in improving performance and should be used. Thus in next section, we will focus on designing even more powerful, but also computationally lightweight feature taking into account both amplitude, frequency and time information.

3.3 Approximate zero-crossing feature

3.3.1 Motivation

Capturing the time component in time-series data is essential. Many features do it through statistical analysis, some through time-frequency analysis. However, many of these features are not easily interpretable. On the other side, when evaluating the EEG/iEEG signals acquired during long-term monitoring, neurologists are capable of identifying prominent ictal patterns among spontaneous EEG/iEEG, noise, and artifacts more accurately than SoA algorithms. In fact, it has been argued [320] that many of the existing ML approaches to biomedical data analysis do not make the effort to integrate the often available expert knowledge into the models. Thus, in this work, we introduce and evaluate a new feature of low-complexity for EEG and iEEG analysis, inspired by the way neurologists visually inspect signals. We call this feature Approximate Zero Crossing (AZC).

Moreover, since ML algorithms tend to present marginal performance gains with the same input data and analysis conditions [321], we hypothesize that the acquisition of high discriminative features such as AZC is an important avenue for improving low-complexity seizure detection methods for resource-constrained wearable devices.

Thus, the main contributions of this chapter are summarized below:

- A new interpretable feature of low-complexity towards resource-constrained wearable devices use, for automated seizure monitoring based on EEG and iEEG is presented. AZC shows a discriminative power higher than most of the other 56 features (CLF) evaluated using the KL divergence in two widely used and publicly available long-term seizure monitoring datasets. Moreover, extracting the AZC features is 8.8x faster than calculating the CLF set of features.
- We propose the time-series-cross-validation (TCSV) methodology for assessing the seizure classification performance, as we believe it presents results that are more realistic for a real-life long-term seizure monitoring application. To the best of our knowledge, it has not yet been used in EEG/iEEG-based seizure detection.
- We show that a ML solution of low-complexity based only on six AZC features achieves 100% sensitivity in 25 out of 42 subjects, plus nine showing only one missing seizure. Moreover, the AZC-based method achieves a low FAR of 2.1 and 1.0/day on average for CHB-MIT and SWEC-ETHZ, respectively.
- We demostrated that the AZC-based classification method can outperform the CLF-based one for seizure detection: 102 and 194 seizures detected versus 99 and 161 for CHB-MIT and SWEC-ETHZ, respectively.

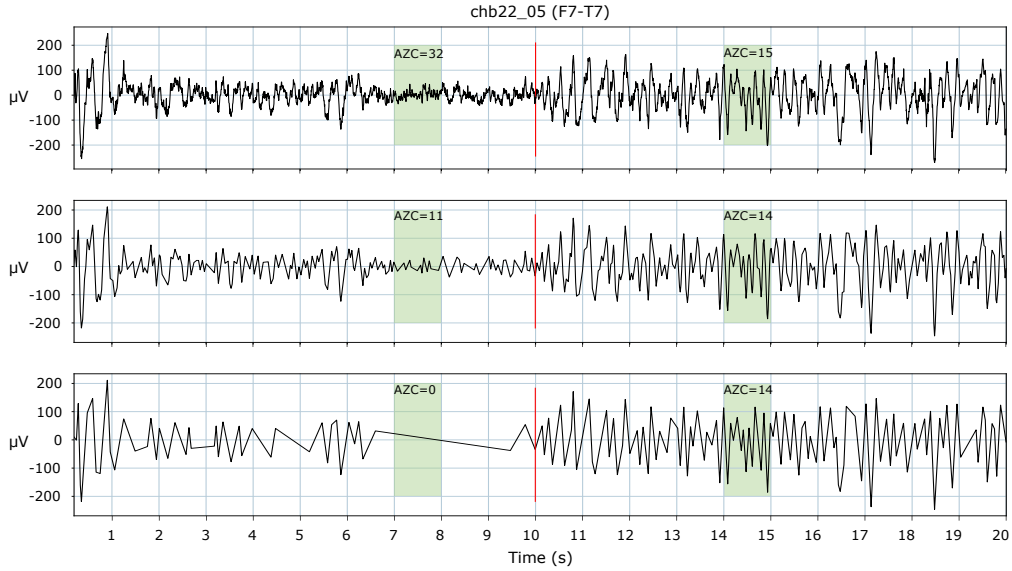

Figure 3.5: Illustrative example of the AZC feature, calculated at different approximation thresholds on a 20 *s* EEG window with a seizure starting at second 10. The top figure is the original signal, while the middle and bottom are approximations with $\epsilon = 16$ and $\epsilon = 64$, respectively. [Source: Patient 22, file 25 from the CHB-MIT dataset.]

3.3.2 Approximate Zero-Crossing

We draw inspiration from clinical practice in EEG/iEEG screening for seizure patterns to develop the proposed AZC feature. The basic idea behind is to estimate the number of peaks in an EEG segment in the same way it is done by a human expert who is visually inspecting the signal. It is therefore implemented as a regular zero-crossing calculation of the first derivative, but before this calculation, the signal is transformed in order to keep only the morphological features that are prominent to the human eye. A family of methods well suited for this type of transformation is a polygonal approximation, also called piece-wise linear representation or linear path simplification [322]. These methods assume that the input signal can be represented as a sequence of linear segments and apply different techniques to obtain the minimum number of segments that best represent the underlying morphology. Formally:

Definition 3.3.1 (Polygonal Approximation) *Given a time series represented as a sequence of n time-value points* $S = \{(t_0, S_0), \ldots, (t_{n-1}, S_{n-1})\}$*, a polygonal approximation of* S *is a subsequence* $L \subseteq S$ *for which the time series behaviour between any pair of consecutive points* $((t_i, S_i), (t_{i+1}, S_{i+1})) \in L$ *is assumed to be linear.*

Under this general definition, we can determine a broad set of non-linear transformations that potentially remove a lot of information from the original signal, depending on how aggressive the approximation is. A coarse approximation will only retain the most outstanding behaviour of the signal, while a fine-grained approximation will remove only the parts that are not even recognizable by the human eye. This level of detail is fully customizable as a parameter of the

approximation transformation.

To illustrate this idea, Figure 3.5 shows an example of a 20-second EEG segment that is approximated using two different thresholds, according to the algorithm described in the next section. Also, we can see how the AZC calculation varies in two different intervals of each signal. In this example, a seizure starts in the second 10, and this is clearly seen in the morphology changes of the signal, with higher amplitude and a more regular frequency. Let us put focus on how the result of the approximation changes before and after the seizure onset. While the approximated signal after this point is almost identical in the three cases (and, therefore, the AZC value is basically the same), the morphology is much more variable before the seizure onset. Indeed, in low-amplitude regions (such as between seconds 7 and 8) AZC varies from a very high value in the original signal, to zero in the case of a relatively high approximation threshold. In the middle figure, we can see that visually the signal is very similar to the original one, but the AZC count in the highlighted segment is less than half. What we want to highlight here is that, while the AZC is fundamentally a feature estimating frequency, its calculation is primarily determined by the amplitude behaviour of the signal. Therefore, it simultaneously captures these two dimensions, which are the ones assessed by the experts during visual inspection.

Approximation Algorithm

There are multiple algorithms to perform the type of morphological approximation we are targeting [322], each of them with different properties, constraints, and computational complexity. Some of them can potentially be implemented in hardware [323] and attached directly to the ADC of the signal sampler, making it possible to directly acquire an approximated signal and, thus, reduce the amount of information to be processed.

Still, the algorithm we selected to perform EEG approximation is the Douglas-Peucker method [324], which requires access to the whole data to guarantee optimality. The main reason for this choice is that the approximation threshold ϵ is directly a measure of amplitude tolerance. Thus, it is fully interpretable, easy to characterize, and lies on the same scale as the EEG signal. Therefore, to ensure that we provide an acceptable approximation for every subject and channel, we propose to simultaneously calculate the AZC feature using different approximation thresholds to cover, in a logarithmic scale, the full range of physiologically meaningful EEG signals.

The operation of the Douglas-Peucker method is detailed in Algorithm 1. The input is a signal fragment S as a sequence of n samples $(S_0, \ldots, S_{n-1})$, and a threshold $\epsilon > 0$ representing the minimum amplitude difference that can be considered to include a point in the approximation. As output, the algorithm returns a sequence $L = (p_0, \ldots, p_{m-1})$, with $2 \leq m \leq n$, representing the samples selected for the approximation. As we can see, the approximation is initialized by the endpoints 0 and $n-1$ (line 3). At each iteration of the outer loop (lines 5 to 16), the approximation is extended with the point of maximum distance to the segment defined by the linear interpolation (INTERP function) between any consecutive pair of points already included

in the approximation (lines 5 and 11). The procedure finishes when none of the points exceeds the minimum amplitude difference (line 6).

Algorithm 1 Douglas-Peucker approximation algorithm

1: **function** DOUGLAS-PEUCKER(S, ϵ)
2: **let** $n = |S|$
3: **let** $P = \{(0, S_0), (n-1, S_{n-1})\}$
4: **let** $S' = \text{INTERP}(S_0, S_{n-1}, n)$
5: **let** $M, k = max(|S_i - S'_i|), i \in [0, n-1]$
6: **while** $M > \epsilon$ **do**
7: $P = P \cup \{(k, S_k)\}$
8: $M = 0$
9: **for all** $j \in [0, \ldots, |P|-2]$ **do**
10: $S' = \text{INTERP}(S_{k_j}, S_{k_{j+1}}, k_{j+1} - k_j)$
11: $M', k' = max(|S_i - S'_{i-k_j}|), i \in [k_j, k_{j+1}]$
12: **if** $M' > M$ **then**
13: $M, k = M', k'$
14: **end if**
15: **end for**
16: **end while**
17: **return** $L = \{(t_p, S_p),\ p \in sorted(P)\}$
18: **end function**

3.3.3 Experimental setup

Datasets

We employ two datasets to evaluate the performance of the proposed features for seizure detection: the CHB-MIT Scalp [242] and the SWEC-ETHZ iEEG datasets [269], as described in detail in Section 2.1.2.

First, we decimated the SWEC-ETHZ data to have the same sampling frequency as the CHB-MIT dataset (i.e., 256 *Hz*). Then, the data of each file was filtered between [1, 20] *Hz* using a zero-phase, 4^{th} order, Butterworth band-pass filter. Thereafter, the filtered EEG data was divided into windows of 4 *s* with a 0.5 *s* step (87.5% overlapping).

Features

We divided feature extraction onto two feature categories:

- **Approximate-Zero Crossing Features (AZC)**: By using different approximation thresholds in the algorithm described in Section 3.3.2, we can obtain highly discriminative features that carry both time and frequency characteristics, and account for intra-/inter-patient variability in seizure patterns (i.e., waveform, amplitude, and frequency) [325]. In this work, we obtain six AZC features. The first feature is calculated without the use of the polynomial approximation, thus representing a classical zero-crossing. The other

five AZC features are obtained by applying the Douglas Peucker algorithm using 16, 32, 64, 128, and 256 μV as approximation thresholds, hence covering the normal EEG amplitude range.

- **Classical Literature Features (CLF)**: We compare the proposed AZC features against a set of features commonly used for EEG-based seizure detection, hereafter denominated CLF features. The used CLF is composed of the 56 features proposed in [36]. From 54 features, 17 are frequency domain features, namely absolute and relative EEG bandpower in various frequency bands: delta [0.5, 4] *Hz*, theta [4, 8] *Hz*, alpha [8, 12] *Hz*, beta [13, 30] *Hz*, and gamma [30, 45] *Hz*. Additionally, we include the total power and the power for [0, 0.1] *Hz*, [0.1, 0.5] *Hz*, and [12, 13] *Hz* bands.
  Then, 37 non-linear features were used: Shannon (SEn), Tsallis (TEn), Rényi (REn), Sample, and Permutation entropies. These entropies are calculated according to [36], [326], considering a histogram with ten bins and adopting $\alpha = 2$ and $\beta = 2$ to calculate Rényi and Tsallis entropies, following [326]. Before calculating all the entropies, we pre-process the data using discrete wavelet transform employing the Daubechies 4 (DB4) basis function. We obtained the approximated and detailed coefficients down to level seven. More specifically, as in [36], sample entropy is calculated from coefficients at levels six and seven; the remaining entropies are calculated from coefficients at levels three, four, five, six, and seven for different input parameters.
  Finally, the two time-domain features were the *line length* as defined in Eq. 3.4, and *mean amplitude* as defined by Eq. 3.5, where $\{S = s_1, s_2, ..., s_n\}$ is a time-series of length N.

$$LL = \frac{1}{N}\sum_{i=1}^{N}(|s_i - s_{i-1}|) \tag{3.4}$$

$$MA = \frac{1}{N}\sum_{i=1}^{N}|s_i| \tag{3.5}$$

Thus, in total, we extracted a total of 62 features (56 CLF + 6 AZC features) per EEG window, per EEG channel. For the CHB-MIT dataset, we performed these steps for the 18 common EEG channels among all subjects. Hence, it yields a total of 108 AZC and 1008 CLF features that are concatenated in matrices in which the lines are equivalent to the number of EEG windows and columns to the total number of features. However, the SWEC-ETHZ metadata does not include information on the iEEG electrode locations. Furthermore, the number of channels among subjects varies greatly from 32 to 128 [269], so we use all available channels for each subject. Finally, we generate labels for the data representing ictal vs. nonictal data. An EEG window is labelled according to its majority class (e.g., a window having 50% or more ictal data is labelled as ictal).

Feature discriminative power

The Kullback-Leibler (KL) divergence is a score widely used in information theory to assess the statistical distance between two probability distributions [327]. The higher the discriminative power, the easier it is to correctly label the data points.

We assess the discriminative power of all features on epileptic seizure detection (i.e., *ictal* vs. *non-ictal*), comparing the KL divergence of the AZC features against features typically used in the literature. We obtain the KL divergence using Eq. (3.6), where *P(x)* and *Q(x)* are the feature probability distributions estimated for *non-ictal* and *ictal* time periods, respectively. Distributions are obtained via a relative frequency histogram drawn from a feature subset $\{X = x_1, x_2, ..., x_n\}$, where histogram bins $\{I_i : i = 1, ..., M\}$ are intervals between X minimum and maximum values. For instance, $P(x)$ is given by the quotient between the number of elements of each bin and X's length.

$$KL(P||Q) = \sum P(x) \cdot \log_2 \frac{P(x)}{Q(x)} \tag{3.6}$$

Machine learning methodology

Targeting a feasible solution for outpatient monitoring, we base our model on the random forest classification algorithm to assess the AZC performance in the seizure detection task. RF is an algorithm based on an ensemble of decision trees to reduce model overfitting. It is fast and lightweight, both in model size and memory footprint [306], and it has been extensively used for EEG-based seizure classification [36], [140], [306]. Thus here we compare epilepsy detection performance using RF with only AZC features and detection performance when using CLF set of features.

Furthermore, as explained in Sec. 3.1.1, we adopt an assessment methodology that represents real-life usage to avoid achieving unrealistic performance metrics. More specifically, we use TCSV cross-validation approach, respecting the temporal relation between data, and avoiding using future data to test on data in the past. We reorganize the extracted feature files to contain the equivalent of one hour of data, no matter the number of seizures available in each one-hour file. Moreover, we guarantee that the first file contains at least one seizure and a minimum of five hours of data, which has been arbitrarily chosen.

As patient-specific detectors can overcome the variability of seizure patterns among subjects, and following previous work [36], [242], we adopt a personalized ML modeling approach.

Performance assessment

Aiming at reducing FAR due to short bursts of rhythmic activity or high amplitude artifacts, we post-process the inferred output's probability of a data window being assigned to *ictal* (*S*) or *non-ictal* (*NS*) classes using Bayesian post-processing, as in detail explained in Sec. 3.1.1. We used window size of 5 seconds (W=10) and a threshold log probability (i.e., classification likelihood CL) of 1.5, according to [280]. Second, we apply a tolerance for seizure detection of 10 sec before and 30 sec after the ground truth when determining the number of true positives

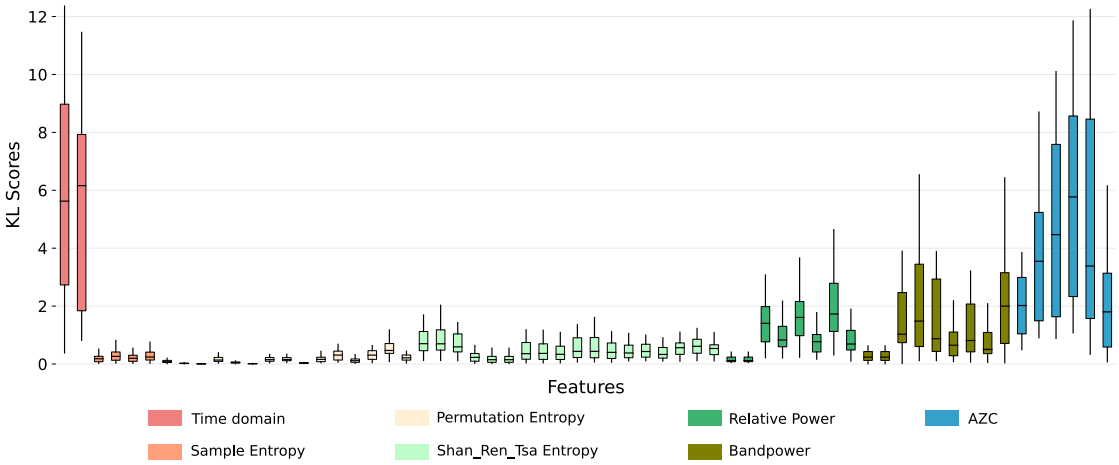

Figure 3.6: KL divergence scores distribution, per feature, for the CHB-MIT dataset. For a detailed description of the features, please refer to Sec. 3.3.3.

and false positives, merging all the true positives that occurred for the duration of a ground truth event. In the end, we compare the post-processed output against the ground truth and measure performance on an episode level using: sensitivity, precision, F1 score, and FAR.

3.3.4 Results

Feature Discriminative Power

We calculate the KL divergence employing histograms with 100 bins. Using box plot visualization of the KL discriminative power among features, we can observe the KL values and also its distribution. We obtain the plot by grouping all subjects' scores per feature, disregarding the EEG channel of origin. Moreover, CHB-MIT is considered more challenging for seizure detection given that scalp EEG acquisitions are prone to suffer from various artifacts (e.g., subject movements, blinking, chewing, electrode disconnections). Consequently, we only present the CHB-MIT dataset box plot of KL scores in Figure 3.6. However, the SWEC-ETHZ iEEG equivalent yields similar results.

The AZC features extracted for different approximation thresholds exhibit KL scores higher than most of the other features. For instance, three out of the five highest median scores are from AZC features obtained with different approximation thresholds (e.g., AZC for 64 μV presents the second-highest median score). Only the *line length* and *mean amplitude* present comparable scores among the 56 CLF features. Meanwhile, features commonly used in SoA works as power features (mainly delta, theta, and alpha bandpower) present relatively smaller scores, followed by Renyi, Tsallis, and Shannon entropies. The remaining features achieved very small discriminative power. In any case, those features with lesser scores are still useful as they can improve separability when combined with others.

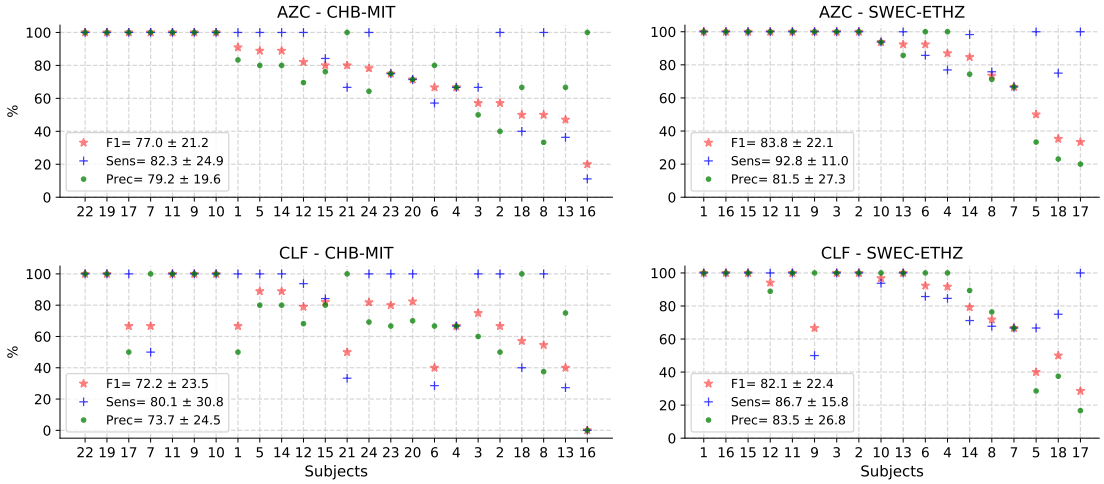

Figure 3.7: Performance metrics obtained with AZC-/CLF-based classification approaches, on both datasets.

Seizure detection

Regarding the seizure detection assessment, Fig. 3.8 and Fig. 3.9 show predictions and true labels in time, using TSCV approach, for randomly selected individual subjects. All available data, except first training set (of duration of five hours or until the end of the first seizure) is concatenated in time.

Table 3.2 and Table 3.3 present the main detection results (i.e., sensitivity and FAR). Overall, we observe that the AZC-based method detected a higher number of seizures compared to the CLF-based one: 102 and 194 against 99 and 161, for CHB-MIT and SWEC-ETHZ, respectively. While, it keeps a similar FAR/day of an average of 2.1 and 1.0 against 2.0 and 0.5, per day, respectively. Moreover, Figure 3.7 shows the F1 score, sensitivity, and precision, per subject, for both datasets and approaches tested (i.e., AZC- versus CLF-based detection). AZC-based detection outperforms the CLF-based one in average performance values for most of the subjects. Overall, 71.4% of patients reached an F1 score greater than 70%, with 84.8% of the tested seizures detected.

Table 3.2: CHB-MIT: number of detected seizures and FAR/day.

Subjects	**01**	**02**	**03**	**04**	**05**	**06**	**07**	**08**	**09**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**	**Tot***
# Seizures	7	3	7	4	5	10	3	5	4	7	3	27	12	8	20	10	3	6	3	8	4	3	7	16	185
# Seiz. Test	5	2	3	3	4	7	2	3	3	6	2	16	11	4	19	9	1	5	2	7	3	2	4	9	132
# Detec. AZC	5	2	2	2	4	4	2	3	3	6	2	16	4	4	16	1	1	2	2	5	2	2	3	9	**102**
# Detec. CLF	5	2	3	2	4	2	1	3	3	6	2	15	3	4	16	0	1	2	2	7	1	2	4	9	99
Data Avail.(h)	41	35	38	156	39	67	67	20	68	50	35	21	33	26	40	19	21	36	30	28	33	31	27	21	979
Data Test (h)	36	20	33	137	33	59	24	15	46	32	2	15	16	21	34	9	16	7	2	18	14	14	21	15	638
FAR/day AZC	0.7	3.6	1.5	0.2	0.7	0.4	0	9.6	0	0	0	11.2	3.0	1.1	3.5	0	0	3.6	0	2.7	0	0	1.1	7.4	**2.1**
FAR/day CLF	3.4	2.4	1.5	0.2	0.7	0.4	0	8.0	0	0	0	11.2	1.5	1.1	2.8	0	1.5	0	0	4.1	0	0	2.3	5.9	2.0

*All the values are a sum for all subjects but the FAR, which are averages.

Table 3.3: SWEC-iEEG: number of detected seizures and FAR/day.

Subjects	**01**	**02**	**03**	**04**	**05**	**06**	**07**	**08**	**09**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**Tot***
# Seizures	2	2	4	14	4	8	4	70	27	17	2	9	7	60	2	5	2	5	244
# Seiz. Test	1	1	3	13	3	7	3	62	24	16	1	8	6	59	1	4	1	4	217
# Detec. AZC	1	1	3	10	3	6	2	47	24	15	1	8	6	58	1	4	1	3	**194**
# Detec. CLF	1	1	3	11	2	6	2	42	12	15	1	8	6	42	1	4	1	3	161
Data Avail.(h)	293	235	158	41	109	146	69	144	41	42	212	191	104	161	196	177	129	205	2652
Data Test (h)	172	5	74	36	103	127	49	126	36	37	29	109	96	73	154	66	46	112	1450
FAR/day AZC	0	0	0	0	1.4	0	0.5	3.6	0	0.6	0	0	0.3	6.5	0	0	2.1	2.1	**1.0**
FAR/day CLF	0	0	0	0	1.2	0	0.5	2.5	0	0	0	0.2	0	1.6	0	0	2.6	1.1	0.5

*All the values are a sum for all subjects but the FAR, which are averages.

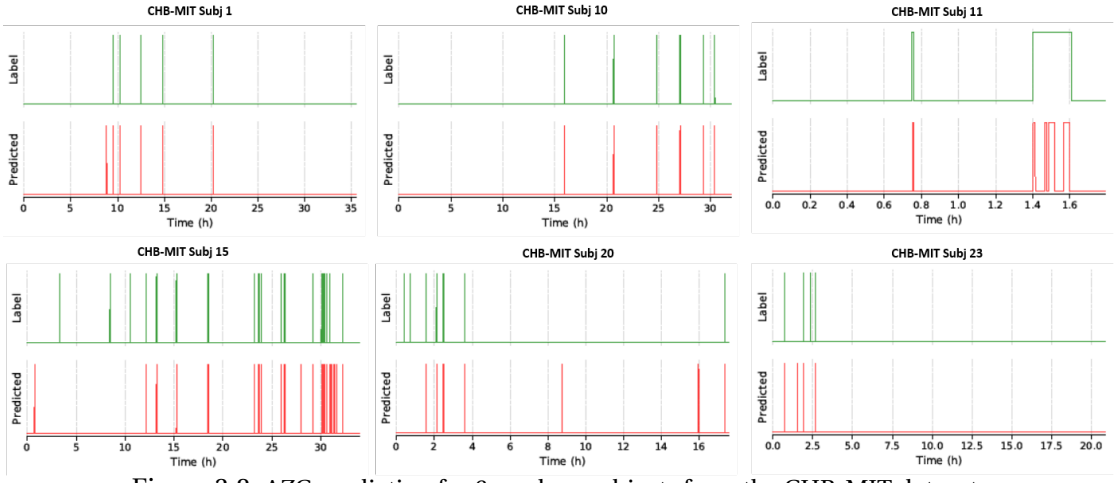

Figure 3.8: AZC prediction for 6 random subjects from the CHB-MIT dataset.

3.3.5 Discussion

The evaluated AZC features achieve higher median KL scores than most of the other features (see Figure 3.6) but also presents a wide range of values. Those values are obtained using all available EEG channels among all patients. Thus, the lack of epileptiform activity in some scalp regions may reduce the separability among data. Moreover, ictal patterns vary greatly among patients, and some patterns can be identified more easily than others [325]. Therefore, AZC could potentially achieve higher discriminative power in case we employ a channel selection approach per patient. Another limitation of our work is related to the set of thresholds used to calculate the AZC values. Regarding seizure patterns difference among patients, a personalized selection of thresholds might also boost AZC's discriminativeness.

Nevertheless, the employed method for extracting AZC features reaches very high performance in both datasets (i.e., 25 subjects out of 42 achieve 100% sensitivity, plus nine showing only one missing seizure). Additionally, we also achieve a very low FAR/day: 2.1 and 1.0/day on average for CHB-MIT and SWEC-iEEG, respectively. These results are achieved using only six features per channel, as opposed to the 56 used in the CLF-based method. Considering the simplicity of the algorithm to calculate the proposed feature (i.e., AZC's calculation is 8.8x faster than the CLF ones and demands less RAM memory), these results support our final goal of using AZC in resource-constrained wearable devices for long-term monitoring of people with epilepsy. Although of low complexity, the AZC feature presents an intrinsic ability to capture changes both in time and in the frequency domain. For example, by approximating the EEG signal under different thresholds, we capture changes in the amplitude of the EEG during an ictal event [328]. Furthermore, the synchronicity of ictal-EEG (that is, slow wave discharges at 3–5 Hz [328]) is also captured by the proposed zero crossing count, while the different thresholds increase the separability of the ictal/non-ictal data.

More importantly, these results were obtained using a particularly demanding TSCV methodology for assessing the classification performance, where models are trained on an incremental

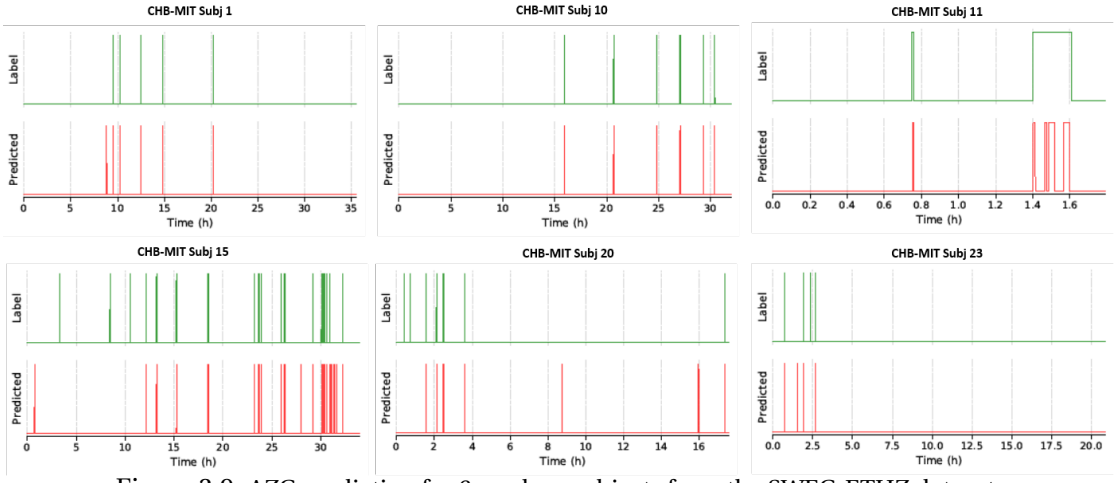

Figure 3.9: AZC prediction for 6 random subjects from the SWEC-ETHZ dataset.

amount of data. To the best of our knowledge, the proposed TSCV approach has not yet been used for assessing seizure detection methods based on EEG/iEEG. Due to its interactive training manner, the TSCV approach makes the assessment harder than other methods commonly found in the literature, and it is not directly comparable in terms of accuracy. Depending on the available amount of seizures per subject and their position in the acquisition timeline, most TSCV-trained models face an even higher data imbalance rate when compared to LOOCV models. The lack of ictal data can cause more misclassifications and false alarms, which was observed for some subjects. However, TSCV maintains chronological data relations, thus providing a more realistic assessment for long-term seizure monitoring approaches.

The LOOCV approach may provide the closest comparison when used to evaluate performance using the whole dataset. For instance, [242] indicates high sensitivity in the CHB-MIT dataset even for subjects declared difficult. By using all seizures but one for training, and by declaring a seizure detection based on only one positive prediction, [242]'s method achieves high sensitivity at the cost of a higher FAR. Although not employing any post-processing, [242] indicates a low average FAR of 3.12/day, which might be explained by the fact that it was only assessed on files without seizures. We report FAR for all tested data, including post-ictal data and in-between sequences of seizures. For instance, by avoiding files containing ictal data, FAR values for *chb12* are not realistic and not comparable to the ones reported in this work (*chb12* has only 11 files with zero seizures out of the 29 currently available). Moreover, this work also reports a relatively high FAR for subject *chb24*, which has not been included in [242].

A similar approach to classification performance evaluation is used by [300] in its patient-specific and hybrid models. The use of fully-connected convolutional neural networks trained in most of the data available, but one record, produced models with high sensitivity to ictal data, but using rather complex architecture-dependant on the tuning of 1000 hyperparameters. Besides the inherent over-optimistic results associated with the LOOCV methodology, as discussed above, by considering only one positive classified window, the minimum need for asserting a seizure detection, [300]'s sensitivity is reported as 100%. Additionally, it also

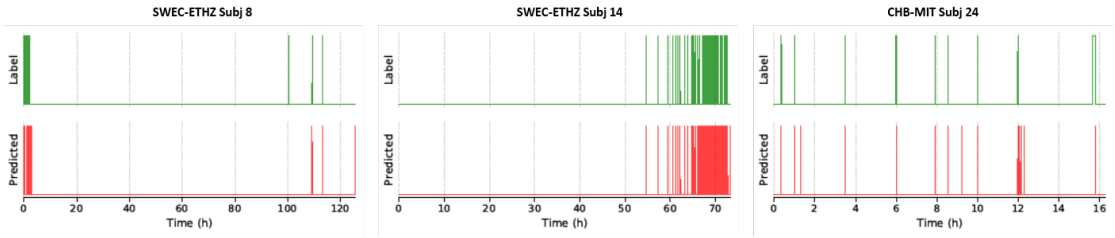

Figure 3.10: AZC prediction for several subjects tackled in the discussion chapter.

reports low average FAR but, once more, calculated only on records not presenting seizures. Thus, [300] did not assess FAR for all available data and, more specifically, it did not employ records with a higher probability of presenting false alarms due to the proximity to true events.

Other works, such as [329], employ a generalized model in their classification approach and are also not directly comparable with our personalized approach. Also using CNN, [329] reports fair metrics for CHB-MIT in a per-window assessment approach (F1 score of 0.59 $\pm$ 0.26) when employing 80% of each subject's first seizures to train its best model. Thus, this work's results corroborate the requirement of a higher amount of ictal data to improve classification performance.

Regarding the SWEC-ETHZ iEEG dataset, the sensitivity we obtained is similar to [269]'s (194 seizures detected out of 217 versus 72 out of 92). However, two remarks are relevant regarding the methodological differences. First, [269] employs the concept of leading seizures for subjects *id08* and *id14*, thus reporting results only for the very first seizure in the long sequence of ictal activity observed in the data of subjects *id08* and *id14* (see Fig. 3.10). This affects sensitivity, as well as, FAR results for both subjects. Considering that those ictal periods are rather long (up to 10 hours for *id14*), we decided to report results for all events. Moreover, [269]'s FAR (average 1.44/day) is considered overestimated as its post-processing is tailored per subject, a rather data-dependent approach, which is also not realistic for real-life applications.

In relation to specific patient results, we would like to draw attention to some patients that are regarded as difficult cases. For instance, patients *chb6* and *chb16* are well known for their short seizures, i.e., 15.3 $\pm$ 2.9 s, and 8.4 $\pm$ 2.3 s, respectively. Thus, they present less ictal data for model training. Nonetheless, the AZC-based detection still performs relatively well for *chb6*. Such performance might be explained by *chb06*'s EEG morphological pattern, which shows a decreasing signal amplitude and increasing frequency during seizures. In such a case, the approximation algorithm associated with the AZC feature calculation approach would further affect the zero-crossing counts, as illustrated in Figure 3.5, thus increasing its discriminative power. Another CHB-MIT particularity, pointed out in [242], is that patient *chb13*'s non-ictal data contains short bursts of rhythmic activity similar to its typical seizure patterns. Hence, it introduces a confounding effect to the proposed algorithm, which uses all available data for training. At last, [325] reported the existence of three unlabeled seizures in *chb24*'s data, immediately after its second-last seizure, which could correspond to the false positives shown in Figure 3.10 around the 11^{th} hour.

Regarding the SWEC iEEG dataset, subjects *id08* and *id14* present a long-lasting cluster of ictal activity labelled as various single events in sequence. As can be seen in Figure 3.10, most of the missed detection and false positives accounted for in both subjects are closer to the cluster boundaries. In a real-life application, such cases would be rather acceptable as detections are close to true ictal events.

Finally, the use of a wide variety of features can boost seizure detection in some cases. For example, the CLF-based methodology detected more seizures than the AZC-based one for some subjects (i.e., *chb03*, *chb20*, *id04*), presenting lower FAR for seven subjects from both datasets, as can be seen in Tables 3.2 and 3.3. These results might be explained by the intrapatient variability in seizure patterns and the presence of various artifacts along the recordings, which requires a fine-grained trained decision boundary to achieve higher performance. However, the AZC-based approach presents results on par for most of the subjects assessed.

3.3.6 Conclusion

In this chapter, we have introduced a new interpretable and highly discriminative feature for long-term monitoring of epilepsy, namely approximate zero-crossing (AZC). Inspired by neurologists' procedures in interpreting and labelling EEG and iEEG data in clinical practice, the AZC feature also targets to mimic how our brain selectively picks prominent patterns among noisy data.

We have employed the KL divergence to show AZC's high discriminative power over a classical set of features used in the SoA. Moreover, we have assessed AZC performance in seizure classification using two publicly available long-term datasets (i.e., CHB-MIT - 982.9 hours, and SWEC-ETHZ - 2656 hours). Overall, an AZC-based classification method detected 102 and 194 seizures against 99 and 161, for the CLF-based one (CHB-MIT and SWEC-ETHZ datasets, respectively), with significantly less features (6 for AZC and 56 for CLF). While, it kept a similar FAR/day, i.e., average of 2.1 and 1.0 against 2.0 and 0.5, per day.

We have also proposed to use time-series-cross-validation approach, an approach more appropriate for long time-series datasets such as epilepsy. It is less prone to overestimated performances reported using leave-one-out cross-validations.

These results showed that there is still space in designing novel features specialized for given applications. We believe that the proposed AZC feature may contribute to developing new methods for outpatient monitoring, particularly on wearable devices, due to its low complexity and high discriminative power.

3.4 Encoding spatial information

3.4.1 Motivation

The last question of this chapter is how to take also spatial information into account. Namely, EEG and iEEG are spatio-temporal, noisy and non-stationary data, whose efficient encoding to HD vectors poses an essential part of the quality of HD learning. However, how to encode channel and feature information has not yet been systematically explored in the literature. In most of the existing literature utilizing EEG or EMG data, only raw data or *local-binary-patterns* (LBPs, [312], have been used as features encoded to vectors. Yet, similarly to standard ML approaches, the possibility to add more features can significantly improve the power of the models, as shown in previous chapters. Thus, in this chapter, we discuss possibilities on how to encode information about channels and various possible features one might need to extract from data, to capture all the relevant aspects.

HD computing has been utilized for epilepsy detection due to its attractive properties, and only with various improvements to standard HD encoding and learning, it can reach the performance of state-of-the-art algorithms. Thus, further exploration on how to optimize HD computing and encoding for better performance is of high interest for epilepsy detection.

Thus, the contributions of this chapter are:

- We discuss not yet addressed topic of optimal encoding of spatio-temporal data, such as EEG, and all information it entails to the HD vectors.
- We propose several approaches to encode together information on features, their values and spatial information about recording channel.
- Epilepsy detection performance is compared as well as memory and computational complexity of each approach.
- We show that including channel information is beneficial for epileptic detection performance, but that the order in which features and channels are mapped to corresponding values is not relevant.

3.4.2 Encoding spatio-temporal data for HD computing

Related work

HD computing has been applied so far in several applications where spatio-temporal data such as EEG, iEEG [204], [240] or EMG [194], [233], [330], [331] was utilized. The majority of works used raw signal values or short local-binary-patterns (LBPs) to describe the changes of the signal in time and map it directly to HD vectors [194], [204], [233], [269], [330], [331]. This enables combining channel and value (raw signal or LBP value) together and then further combining all samples belonging to the same class. Some authors have also encoded time information between neighboring samples by utilizing vector permutation. Furthermore, in [332] authors propose energy-efficient in-memory encoding for this way of encoding spatio-

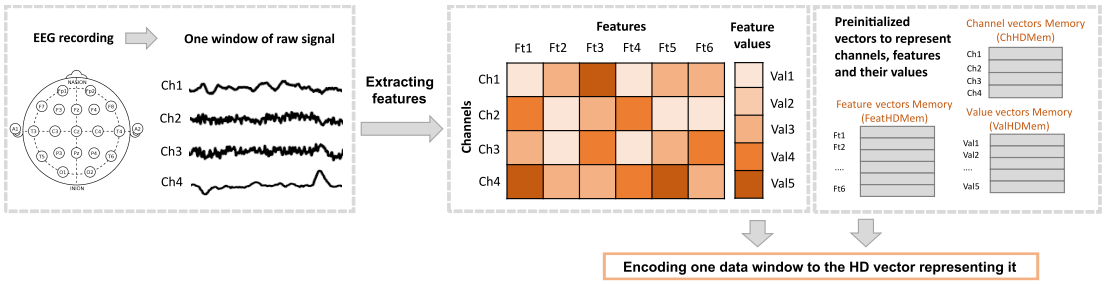

Figure 3.11: Illustration of encoding one data window to HD vector representing it.

temporal signals. This is an interesting approach in its simplicity as it use directly raw signal values. However, it is limited to only mapping one type of information to HD vectors (i.e., raw signal amplitude or/and signal change trends).

In the previous chapter, we tested how different feature types regularly used in non-HD ML classifiers for epilepsy detection perform in the HD computing framework. It was shown that utilizing more different statistical, time, or frequency features can outperform simple LBP or raw signal encoding. Further, in recent [240], authors extend their previous work [269] by adding mean amplitude and line length features to LPB values. They resolved the problem of encoding more features to HD vectors by having three independent classifiers, each with its own model vectors. Predictions are merged using a simple linear layer that gives a final prediction based on distances from class models of each feature. This is a promising approach that enables the performance comparison for the different features.

In [333], authors explored epileptic seizure detection using power-spectral features from iEEG data, and encountered similar problems as we point out here. They explored three different encoding approaches: 1) concatenating feature HD vectors to generate long HD vectors, 2) using multiple classifiers (one for each feature) and then integrating their predictions by majority voting, and 3) training one classifier using all features. In the last approach, vectors representing features and their values are combined in the first place and then united with channel information. These approaches are highly interesting due to the broad spectrum of feature characteristics that are integrated. However, unfortunately, in [333] they were not systematically compared from performance, memory or complexity aspects, thus focus on that more thoroughly here.

Lastly, two recent papers discuss encoding graph data into HD computing framework [222], [334]. In [222], an encoding method representing complex graph structure while supporting both weighted and unweighted graphs was proposed. The authors of [334], compared to the state-of-the-art Graph Neural Networks (GNNs) the proposed graph classification model and achieved comparable accuracy, with faster training and inference time.

Proposed approach

Typical spatio-temporal data, such as EMG or EEG, is 2D data. It consists of several channels positioned in different physical locations (which can have specific relations between them) and

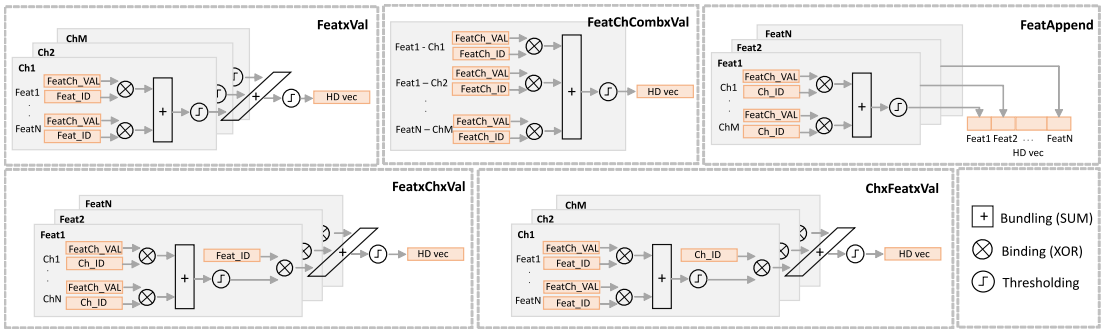

Figure 3.12: Schematic of different possibilities to encode information about channels, features and their values to HD vectors. Five different encoding approaches tested in the paper are illustrated.

recorded during time frames of various lengths. After preprocessing the signals, features are typically extracted in time, using time windows shifted by some time step. As this is repeated for each of the channels, this leads to 3D data containing information on the feature, channel, and time, as illustrated in Fig. 3.11. Within the HD computing workflow, this means that we need to define initial vectors that represent each of these entities: HD vectors representing each feature ($Feat_{ID}$ as in Fig. 3.11), and vectors representing each possible value of features ($FeatCh_{VAL}$), and channels (Ch_{ID}). As values of various features might not have the same range and are not usually integer values, they need to be normalized and discretized to a specific number of bins.

The question that we address in this chapter is how to encode all this information into HD vectors. Moreover, we assess the different possible methods from several perspectives, which include classification performance, memory and computational complexity. The last two metrics are very relevant in the design of the next generation of smart wearable systems. Therefore, in Fig. 3.12 we illustrate different possibilities considered to encode the time window of data to an HD vector. The time window contains information on all features and their values calculated from each of the channels.

A typical approach dealing with simpler, non-spatial data is to bind feature vectors with feature value vectors and then bundle (and threshold) them. Translating this approach to EEG data leads to two options: 1) $Feat \times Val$ and 2) $ChFeatComb \times Val$. The difference between them is, as illustrated in Fig. 3.12 and formulated by (3.7 and 3.8), that $Feat \times Val$ does not include information about channels but simply bundles (and normalizes) features ($Feat_{ID}$) and feature values from each channel ($FeatCh_{VAL}$). On the other side, $ChFeatComb \times Val$ approach treats each feature and channel combination as an individual feature and gives it an independent HD vector ($FeatCh_{ID}$). This approach distinguishes between channels as formulated in (3.8), but can lead to significantly large memory maps to store all initial HD vectors. This could happen in case data contains many channels (such as iEEG data) or when many features are extracted from each channel.

As this can be problematic from a memory perspective viewpoint in the context of wearable medical devices, we propose other more EEG-inspired approaches: 3) $Feat \times Ch \times Val$ and

4) $Ch \times Feat \times Val$. Both approaches first initialize vectors for each channel (Ch_{ID}), each feature ($Feat_{ID}$), and possible feature values ($FeatCh_{VAL}$). The difference is, in the order of bundling information. For $Feat \times Ch \times Val$, as formulated by (3.9), in the first step, channels (Ch_{ID}) and feature values on that channels ($FeatCh_{VAL}$) are bundled. Then, in the second step, the bundling with feature ID's ($Feat_{ID}$) follows. In $Ch \times Feat \times Val$ approach order of bundling is opposite, as formulated in (3.10).

$$Feat \times Val = \left\lfloor \sum_{i=1}^{numCh*numFeat} Feat_{IDi} \oplus FeatCh_{VALi} \right\rfloor \tag{3.7}$$

$$ChFeatComb \times Val = \left\lfloor \sum_{i=1}^{numCh*numFeat} FeatCh_{IDi} \oplus FeatCh_{VALi} \right\rfloor \tag{3.8}$$

$$Feat \times Ch \times Val = \left\lfloor \sum_{i=1}^{numFeat} Feat_{IDi} \oplus \left\lfloor \sum_{j=1}^{numCh} Ch_{IDj} \oplus FeatCh_{VALij} \right\rfloor \right\rfloor \tag{3.9}$$

$$Ch \times Feat \times Val = \left\lfloor \sum_{i=1}^{numCh} Ch_{IDi} \oplus \left\lfloor \sum_{j=1}^{numFeat} Feat_{IDj} \oplus FeatCh_{VALij} \right\rfloor \right\rfloor \tag{3.10}$$

The last approach, called $FeatAppend$, is designed with the goal to make it easier to interpret encoded vectors and determine which part of them comes from different features. In this approach, as illustrated in Fig. 3.12, channels and feature values are bound, bundled, and thresholded to get a vector representing the encoded sub-vector for each feature. Instead of binding it with other feature sub-vectors as in $Feat \times Ch \times Val$ these vectors are appended one after another, as formulated in (3.11). In order to get the same final vector dimension, initialized sub-vectors have, in this case, smaller dimensions. This encoding organization enables to analyze vectors for every single feature which will be used at the end of this thesis, in Chapter 6.3, for performing feature selection.

$$FeatAppend = \left\lfloor \sum_{j=1}^{numCh} Ch_{IDj} \oplus Feat_1 Ch_{VALj} \right\rfloor ... \left\lfloor \sum_{j=1}^{numCh} Ch_{IDj} \oplus Feat_{numFeat} Ch_{VALj} \right\rfloor \tag{3.11}$$

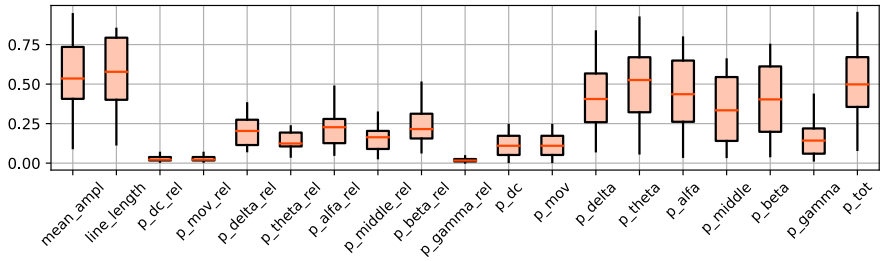

Figure 3.13: Jensen-Shannon divergence of features.

3.4.3 Experimental setup

Dataset

For the analysis, we use the CHB-MIT database as described in Sec. 2.1. Even if the common approach in literature is using balanced data preparation, it can lead to highly overestimated performance as will be shown in next chapters. Further, training on the full dataset using HD computing is very time consuming. Thus, we use a data selection approach that contains all seizure segments and ten times more non-seizure data. Data is arranged in such a way that for each seizure file, seizure data is extracted and surrounded by non-seizure data, which was randomly selected from one of the files not containing any seizure. In this way, each file contains an equal ratio of seizure and non-seizure data, and can be easily used for leave-one-file-out cross-validation.

Feature Choices

In the previous chapter 45 features classical literature features were used for classification, including mean-amplitude, 37 entropy features and 8 relative frequency domain features. Based on a literature review by [140] discussing the importance of various features, frequency features, as well as line-length were reportedly the most useful. Thus, in this paper, we keep the mean-amplitude ($mean\_ampl$) and use both relative and absolute values of power spectral density in the five common brain wave frequency bands: delta: [0.5-4] Hz (p_delta and p_delta_rel), theta: [4-8] Hz (p_theta and p_theta_rel), alpha: [8-12] Hz (p_alpha and p_alpha_rel), beta: [12-30] Hz (p_beta and p_beta_rel), gamma: [30-45] Hz (p_gamma and p_gamma_rel), and low-frequency components: [0-0.5] Hz (p_dc and p_dc_rel) and [0.1-0.5] Hz (p_mov and p_mov_rel). To calculate the relative values, we divide the absolute values by the total power (p_tot). In this case, we do not to use entropy features, as a preliminary analysis and the literature showed a very small discriminative power. In the end, we included the line-length feature ($line\_length$), as introduced in [96], and which showed a high discriminative power in the preliminary analysis. Thus, in total, we extract 19 features. Before extracting the features, the data is filtered with a zero-phase, 4th-order Butterworth band-pass filter between [1, 20] Hz. Features are extracted from data segmented into 4-second windows with a 0.5-second step.

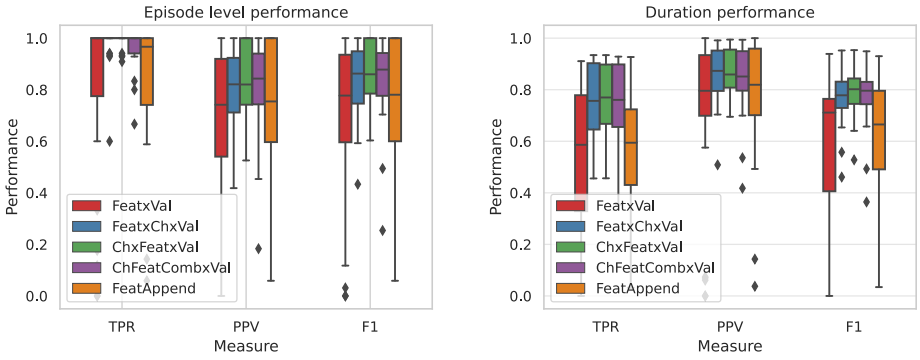

Figure 3.14: Performance of different EEG encoding approaches. Performances on the level of seizure episodes and seizure duration are shown. TPR or true positive ratio is the measure of sensitivity, PPV or positive predictive value is the measure of precision, and F1 score is gmean between TPR and PPV.

HD Learning Workflow

The standard HD computing workflow consists of a single-pass training, where HD vectors representing different data windows (in this case, four-second windows) from the same label class are bundled all together to form a model HD vector representing that class. This approach is simple and fast. However, all data windows are equally important, which can, in highly imbalanced datasets such as epilepsy ones, lead to the domination of more common patterns in the final vectors. Thus, here we test 'OnlineHD', which was proposed in [335] as an alternative approach, where each window vector is multiplied with a weight before being added to the model vector. 'OnlineHD' has been described in more detail in Sec. 1.3.2.

Performance evaluation

Training and evaluation are done independently for each individual due to the subject-specific nature of epileptic seizures. For each person, we perform leave-one-seizure out cross-validation and report the performance as the average across all cross-validation runs. In the end, we report performance as average over all subjects.

The performance of the classifier is evaluated with respect to seizure episode detection and duration-based (window-based) detection. More specifically, we measure sensitivity, predictivity, and F1 score on the level of episode detection, as well as on the level of episode duration. Finally, to have one single final metric, we combine F1 scores for episodes ($F1E$ and $F1D$) using the geometric mean as $F1DEgmean$. We also post-process raw label predictions by performing a moving average with majority voting, using a window size of 5s.

Finally, the code and data required to reproduce the presented results are available online as open-source (link will be available after the double-blind peer-review process).

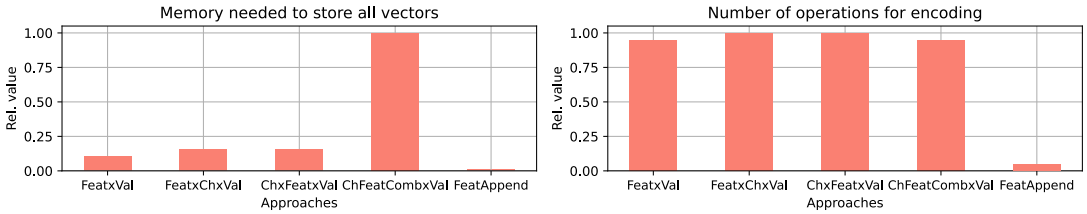

Figure 3.15: Comparison of encoding approaches in terms of memory needed and number of binding and bundling operations needed to encode one data window.

3.4.4 Results

Feature comparison

Fig. 3.13 shows Jensen-Shannon divergence of 19 features we used and their distribution over all channels for all subjects. There is a clear difference between features, where mean amplitude, line length, total energy, and absolute spectral powers of the delta, theta, alpha, beta, and middle-range are quite discriminative. Relative powers seem to be less discriminative than absolute values.

Epilepsy detection comparison

On Fig. 3.14 we compare the average seizures detection performance for all subjects without any post-processing. $Feat \times Val$ encoding, which does not takes channel information into account, leads to lower performance encoding types that take channel information into account. There is no significant difference in performance between the three approaches that include channel information: $Feat \times Ch \times Val$, $Ch \times Feat \times Val$ and $ChFeatComb \times Val$. The $FeatAppend$ approach, even if including channel information, yields a smaller performance than the three approaches that utilize channel information, probably due to the smaller number of dimensions per feature. Yet, performance is not worse than the $Feat \times Val$ approach.

Memory and computational complexity

Fig. 3.15 shows the memory needed to store all HD vectors (for channels, features, and values) for each of the approaches is shown. Relative ratios are shown as memory directly depends on the chosen dimension D (in this case 19000) and whether HD vectors are binary or not. Due to the large number of combinations of features and channels, the $ChFeatComb \times Val$ approach is the most memory-demanding one, and scales the worst. $FeatAppend$ approach is the approach that requires the least amount of memory as base vectors have lower $D/numFeat$ dimensions.

Further, the relative number of bundling and binding operations needed to encode one window of data to the HD vector representing it is shown as well in Fig. 3.15. The $FeatAppend$ approach requires significantly less operations due to the effectively smaller number of dimensions per vector for each feature. From other approaches, $ChFeatComb \times Val$ and $Feat \times Val$ require slightly less operations then $Feat \times Ch \times Val$ anr $Ch \times Feat \times Val$.

3.4.5 Conclusion

This chapter draws attention to not yet discussed and properly explored topics of mapping and encoding spatio-temporal data such as EEG or EMG to HD vectors. Although HD computing has been utilized and has shown promising results for various biomedical applications (in particular utilizing EEG and EMG), most works in the literature only use raw data (or LBP values), namely, only one feature per channel. Thus, the optimal encoding when more features per channel or more available data modalities are used remained unclear.

In this section, we propose five alternatives to encode feature values for all channels of one data window into an HD vector and test it on epileptic seizure detection. Our results show that including channel information is beneficial for epileptic detection performance, but that the order in which features and channels are mapped to corresponding values is not relevant. Further, we show that the $ChFeatComb \times Val$ approach is the most memory demanding and that $Feat \times Ch \times Val$, $ChxFeat \times Val$ and $FeatAppend$ are comparable and more appropriate. $FeatAppend$ requires the least amount of memory and operations to encode vectors due to the effectively smaller number of dimensions per feature vector. By creating individual vectors for each feature and appending them next to each other, in $FeatAppend$ performance is slightly reduced. This situation is probably due to fewer HD dimensions used per feature. Thus, it could be improved by increasing the total dimension of vectors, but this also leads to higher memory requirements, which could make it less friendly for wearable devices.

3.5 Conclusion

This chapter focused on setting a baseline on how to encode EEG data to HD vectors. First, we explore whether it is beneficial to extract epilepsy-specific features, or rather use just raw signals as in literature, while employing HD computing for epilepsy detection. The question was which and how many features we should encode. Next, we also designed a new highly discriminative feature that simultaneously encodes time and amplitude information. In the end, we explored how to also include spatial information during the mapping process. This has not yet been discussed in the literature, and we hypothesized that including this information could improve performance.

In more detail, in Sec. 3.2, we performed a systematic assessment of feature encoding strategies for the detection of epileptic seizures using HD computing framework. More precisely, we tested two known feature methodologies (*LBP* and *RawAmpl*), as well as several novel approaches for HD computing on epileptic seizure detection (Single feature - *Ampl*, *Entr*, *CWT*; Multiple Feature - *3Feat* and *45Feat*, and *FFT*). We compared three main aspects: 1) detection performance, 2) memory requirements, and 3) computational complexity. Results show a different performance between approaches, that amplitude is a very valuable feature, but that also using more features helps. For memory and computation reasons, the highest performance approaches (*45Feat* and *RawAmpl*) are not ideal for wearable applications due to their high memory and computational requirements. Results also showed that the differences between approaches are much smaller concerning memory than computational complexity. Nonetheless, after post-processing steps for smoothing prediction labels to adjust predictions to the dynamics of epileptic seizures, performance is improved, and differences are significantly reduced between approaches. Overall, these results point in two directions. Using only LBP or raw amplitude features is not necessarily ideal, and using more epilepsy discriminant features can be useful both from performance, memory, and computational aspects. And secondly, for time-series data, post-processing plays a big role in improving performance and should be used.

Then, in Sec. 3.3, we have introduced a new interpretable and highly discriminative feature for long-term monitoring of epilepsy, namely approximate zero-crossing (AZC). Inspired by neurologists' procedures in interpreting and labeling EEG and iEEG data in clinical practice, the AZC feature also targets to mimic how our brain selectively picks prominent patterns among noisy data. We demonstrated AZC's high discriminative power over a classical set of features used in the SoA. Moreover, we have assessed AZC performance in seizure classification using two publicly available long-term datasets and showed superior detection using just 6 AZC features when compared to 56 classical literature features. These results showed that there is still space in designing novel features specialized for given applications. We believe that the proposed AZC feature may contribute to developing new methods for outpatient monitoring, particularly on wearable devices, due to its low complexity and high discriminative power.

In the last section of this chapter, Sec. 3.4, we explore topics of mapping and encoding spatio-

temporal data such as EEG or EMG to HD vectors. We have proposed five ways to encode feature values for all channels of one data window into an HD vector and have tested them on epileptic seizure detection. Results showed that, indeed, including channel information is beneficial for detection performance. It is interesting to see that the order in which features and channels are mapped is not relevant. We further compared approaches in terms of required memory and also computational complexity. $ChFeatComb \times Val$ approach is the most memory demanding, while $Feat \times Ch \times Val$, $ChxFeat \times Val$ and $FeatAppend$ are comparable and more appropriate. In the end, $FeatAppend$ approach, requires the least amount of memory and operations to encode vectors due to the effectively smaller number of dimensions per feature vector, and it is also interesting for the possibilities to compare performances and confidences per individual feature. This will be utilized in the last chapter of the thesis, Chapter 6, for feature and channel selection, as well as visualization of predictions per channel and feature in time.

Limitations

As the majority of works dealing with large and unbalanced databases, such as epilepsy ones, work in chapter 3.2 and chapter 3.4 was done only on a sub-selection of the of CHB-MIT database due to computational limitations. Namely, training on the whole database using HD computing took an extensive amount of time. Thus, by using a random selection of non-seizure data and the same amount 10 times more non-seizure than seizure data, we tried to minimize the negative effects of not using the whole database. Thus, later in the thesis, 1) we tried to quantify how sensitive is HD computing to subselection and unbalance in dataset, and 2) we worked on adapting the code for GPU processors to be able to reproduce such analysis with the whole database.

The encoding approach based on feature appending is simple and enables a detailed comparison of each feature, but it has the drawback that the dimension of the HD vectors scales with the number of features, potentially making it more memory and time-consuming. At the same time, reducing the number of dimensions per feature lowers the information capacity that can be stored and ultimately lowers the performance. Thus, different approaches that enable the comparison of a high number of features without limiting the number of dimensions per feature should be designed in the future.

Furthermore, we introduced the topic of vector initialization, but we did not research it in depth. Thus, there is research to be made on the optimal initialization of feature and channel base vectors. More specifically, if the spatial location of EEG electrodes is known (such as for a 10-20 scalp EEG system), EEG-based initialization that will map 2D/3D distances between channels to the similarities between vectors should be tested. The hypothesis is that such initialization could further improve performance. For iEEG datasets, iEEG contact locations are known by the physicians, thanks to a post-implantation CT or MRI scan, and they usually respect a spatial ordering (for instance, several contacts linearly spaced on the same shaft) that could be leveraged. However, this information is often kept out of public datasets, and in this case of an unknown exact location, the only option is random initialization.

4 Learning with unbalanced datasets

4.1 Dealing with unbalanced and large datasets

Class imbalance is one of the major problems in machine learning [251], and the majority is seen in medical datasets [252]. Epilepsy, and as such also epilepsy datasets, are highly imbalanced datasets, i.e., they have a significantly higher portion of inter-ictal (non-seizure) data when compared to ictal (seizure) data [253].

Machine learning models (e.g., neural networks and support vector machines) can be very susceptible to class imbalance [336], [337] and this has to be taken into account. For example, performance can be very poor if training on highly imbalanced datasets, whereas significantly better if a balanced subset is used or if some data augmentation methods are used. For example, in [338] authors used the under-sampling method with the SVM classifier, and in [339] authors used a combination of SMOTE and random under-sampling with boosting techniques. The problem with using data augmentation is that the minority class might still not be fully represented.

Oppositely, when creating a selection of the majority class from unbalanced dataset to get a (semi) balanced subset, the majority class can be underrepresented. This can lead to very good performance results on selected data subsets [128], [340], but as it does not represent the true nature of epilepsy data, these results could be highly overestimating results on the original data. More precisely, any results not done on the original dataset have to be taken with caution, as this is not realistic for real online applications of the same algorithm.

Boosting or bagging techniques that consist of an ensemble of weak classifiers or cost-sensitive models that consider the misclassification cost of each class will often perform better on datasets that are imbalanced [251], [341]. Furthermore, some researchers focused on improving the learning of neural networks in the case of unbalanced datasets. For example, authors of [342] proposed an imbalance learning approach with a weighted cost function for ANN to improve the accuracy of the highly imbalanced CHB-MIT seizure dataset. Similarly, in [343], a novel method based on the weighted extreme learning machine was proposed.

However, for HD computing applications, almost no discussion exists in the literature related to dealing with highly unbalanced datasets. When talking about epilepsy, to the best of our knowledge, HD computing has not been tested on large datasets such as those for epilepsy detection. Thus, in this chapter, we first discuss methodological aspects that must be considered when dealing with highly unbalanced datasets. Then we discuss smarter learning approaches and propose a 'multi-centroid' approach to tackle the unbalance problem for datasets with also high variability of data. In the end, we propose an open-source, PyTorch-based library built for speeding up HD training and easier exploration of the HDC algorithms.

4.1.1 Methods

The first step when using large and highly unbalanced datasets is to consider the methodological choices to make. As already discussed in Sec. 2.3, there is a broad range of decisions to make. We need to decide if there is a possibility to use the whole unbalanced dataset for training and testing. This would be ideal, but it is also often not possible. For example, some algorithms require too much computational or memory power, more than we can have in our setup. Sometimes even if we could run the training, the time it would take is so long that we could not properly work on optimizing the parameters of the algorithm. Thus, in this case, we have to resort to creating a subsection of the original data. In cases when one class is in the minority, it is usually possible to use all data samples of that class and then select a subset of the majority class.

In the case of epilepsy, and in this thesis, we resorted to using data subsets, whereas a parameter, we change how much more non-seizure data is kept when compared to non-seizure data. More specifically, all seizure data is used, and we randomly select non-seizure data from the same subject to construct 'new files' that contain data. We tested three different data distributions: 1) a balanced case with an equal amount of seizure and non-seizure data, and 2) and 3) unbalanced with 5 and 10 times more non-seizure data. We call these three distributions F1, F5, and F10, respectively. Bigger unbalance was not possible due to not enough non-seizure data for all subjects. Later in section 4.2.4, we analyze the performance of these different data subsets. Another option is to test other more complex undersampling and oversampling techniques.

In case it is possible to use the whole dataset, a decision on data segmentation is necessary. It also overlaps with the decision on train/test split and cross-validations approach. As in chapter 3, we explained that there is also important information in the time component for time series data. Thus, randomly shuffling data samples and splitting them into train and test sets is not appropriate. On one side, prediction results can be significantly improved by postprocessing predictions taking into account inherent data distribution. On the other side, randomly shuffling samples might result in having two samples that are on a few seconds apart into the train and test set. This can significantly bias performance results. Thus we suggest that data is treated as a time series instead of treating it as sample-by-sample.

In the end, metrics that are used to evaluate the performance of the algorithms might play even the most important role. Namely, it is clear that in highly unbalanced datasets, accuracy itself is not a very representable measure. With 90% of the majority class in the training and 90% accuracy, it might mean that model classifies everything as the majority class. Thus, for this reason, it is necessary to use at least two metrics like sensitivity and specificity or more complex metrics like F1-score, balanced accuracy, or ROC curve. But even they can also be hard to interpret and even misleading. Thus in the case of the detection problem where classes are time grouped into 'events', even-based metrics have been proposed. We introduced and discussed this in section 2.3.5. Thus here, we also use these performance metrics; event/episode based and also sample-by-sample/duration based metrics.

4.1.2 Smarter learning approaches for HD computing

Another way to take the unbalanced nature of data into account is to design smarter learning approaches. In the literature, there have been few approaches to improve the learning capabilities of HD computing systems.

Single-pass learning does not always leads to satisfying accuracy. Thus, iterative training is a logical potential solution. However, a lack of controllability of training iterations in HD classification can also result in slow training or even divergence. To solve this training issue [227] proposed a retraining approach called 'AdaptHD'. The authors introduced the definition of learning rate in HD computing. Two retraining approaches are tested: iteration-dependent and data-dependent. In the iteration-dependent approach, 'AdaptHD' changes the learning rate based on the changes in the classification accuracy during retraining, making it big at the beginning and lowering as performance improvement gets smaller. In a data-dependent approach, the learning rate is adapted for each data point, depending on how far off the data was misclassified. Finally, the authors also tested the hybrid approach, which updates the learning rate considering both iteration and data dependency. Four different applications and datasets were used to test approaches: Speech Recognition (ISO-LET) [344], Activity Recognition (UCIHAR) [345], Face Recognition (FACE) [346], and Cardiotocography (CARDIO) [347]. It turned out that optimal learning rates are highly specific to a dataset and must be optimized. Namely, too small learning rates need too many iterations, which is costly, and too big learning rates can miss an optimal performance and not converge. However, iterative training did lead to a slightly better detection performance.

Iterative training can boost performance but it also loses some advantages of single-pass training: it is more computationally and time consuming. Thus, the idea of not treating all samples as equally important but multiplying them with a certain weight before adding to the model has been further extended, and called 'OnlineHD' [192]. As explained in 1.3.2, new data vectors are multiplied by their similarity to current class vectors before being accumulated, as illustrated in Equations 1.3 and 1.4. Difference is that instead of retraining models, 'OnlineHD' trains in a single-pass, but identifies common patterns and eliminates model saturation by

more common class. Namely, naively accumulating all samples can cause forgetting and saturation in each class hypervector, where the pattern of common data dominates each class [192]. Authors show in [192] that 'OnlineHD' adaptive training provided, on average, 12% higher classification accuracy on six different datasets when compared to single-pass training, while getting all efficiency benefits that a single-pass training provides.

In this thesis, we, however, propose another approach. Namely, the problem of saturating by more common patterns can be potentially avoided by allowing more than one model vector per class. We call this approach 'multi-centroid' approach and test it on the use case of epilepsy detection. Epilepsy is a good use case since: 1) seizures show highly personalized patterns that are not necessarily constant even within one subject (will be more discussed in Section 5, and 2) non-seizure data, represents recordings form many different physiological states (sleeping, thinking, physical activity, etc.) that do not necessarily have the same brain signatures. In the next section Sec. 4.2, we describe in detail the proposed 'multi-centroid' approach, and then we systematically compare it in Sec. 4.3 with 'OnlineHD' and retraining approaches from the literature.

In the end, as mentioned, to the best of our knowledge, HD computing has not been tested on large datasets such as those for epilepsy detection. Since in chapter 4.2 we show that HD computing, similar to other models, can overestimate performance on smaller subsets of data, we find it important to be able to train models on whole datasets. But unfortunately, training on such large datasets is challenging from a computational perspective. For that reason, in Sec. 4.4, we propose HDTorch, an open-source, PyTorch-based library built for speeding up HD training. Such libraries can enable easier exploration and adaptation of the HD computing paradigm for large unbalanced problems.

4.2 Multi-centroid learning

4.2.1 Motivation

Traditionally, HD computing classifiers have been based on creating one model vector for each target class. However, a challenging aspect of electroencephalogram (EEG) signatures of epileptic seizures is their uniqueness and high variability among people, brain states, and time instances, especially if they are grouped under only two given labels (seizure and non-seizure). Further, non-seizure data also contains many different brain states, such as awake, sleeping, physical or mental effort conditions, etc. All of these states have their own brain signatures. Thus, we hypothesize in this work that creating multiple sub-types (model vector centroids) of seizure and non-seizure classes, based on both labels provided by a neurologist and also on EEG signal characteristics, can be beneficial for learning and prediction. We called this approach "multi-centroid", as we allow to have more vectors (centroids) representing sub-types of each class. This is a form of semi-supervised learning, as the main labels (seizure or non-seizure) are known, but an unrestricted number of seizure (and non-seizure) sub-types is created during training.

More precisely, in this work, we contribute to state of the art in the following manner:

- We design a semi-supervised HD computing approach of learning based on the unconstrained creation of several prototype vectors/sub-classes (unlabeled) of main (labeled) classes.
- We implement this novel approach for epileptic seizure detection based on EEG signal recordings, leading to the creation of multiple prototype vectors/sub-classes for seizure and non-seizure. We evaluate and show a significant improvement in the performance when compared to the standard 2-class HD approach.
- Since a high number of prototype vectors penalizes memory efficiency, we designed two algorithms to reduce the number of sub-classes in the post-training stage. One is based on removing less populated sub-classes, and the other is based on clustering of sub-classes.
- We measure the performance improvement of this approach and analyze the number and structure of sub-classes based on the publicly available CHB-MIT epilepsy database. We show that this approach has greater improvements for more unbalanced datasets while not significantly increasing the number of sub-classes compared to balanced datasets. Thus, this multi-centroid approach can be an essential element to achieve high performance of epilepsy detection with real-life data structures. Moreover, it can be particularly relevant during online learning, where seizures are infrequent.

4.2.2 Multi-centroid HD training

Related work

In the literature, few papers are applying different semi-supervised learning and clustering approaches to HD computing. In [191], authors allow iterative expanding of the training data by labeling unlabeled data points, which can be classified with high confidence by the current model. This improves the quality of prediction by 10.2% on average on 18 popular datasets. This approach can be highly beneficial for epilepsy due to the high amount of unlabeled data that can be accumulated during patient monitoring but cannot be fully labeled by the experts. A potential problem is that it would strengthen common patterns, but could under-represent less common patterns.

In [348], the k-means algorithm is adapted to the HD computing paradigm. This means that, before clustering, data is mapped to HD vectors. Then, properties of HD vectors are used to perform clustering on a preset number of classes. The authors compared it with k-means on 9 different datasets, and the influence of various parameters was investigated. Results showed the same or better performance than the standard non-HD k-means algorithm for all datasets. The disadvantage of this approach is that the number of sub-classes has to be preset, which can be quite challenging in the case of an epileptic seizures. In fact, this number would be different for every patient and it may even change in time as more training data is added. Further, it does not use the information about the global (seizure/non-seizure) labels that are available.

Another approach for semi-supervised learning is the idea of relearning, in which the algorithm iteratively passes through the training set. In the case of a mis-classification, the sample is removed from the mis-classified class and added again to the prototype vector of the correct class. Therefore, iterative learning tries to overcome the problem of single-pass learning, i.e. that it can lead to the saturation of the prototype vectors of each class by data that are more common in each class and perform badly on under-represented patterns of the same class. In [227], authors tested iterative approaches with different fixed and adaptive learning rates on several datasets for speeding up learning and saving energy while keeping the same or higher accuracy as single-pass training.

In [349], the authors targeted to achieve the higher performance of iterative training while keeping the speed and simplicity of single-pass training. The approach, called 'OnlineHD', is single-pass, but adjusts the weight of each example according to the similarity with the trained prototype vectors. This leads to an accuracy increase of 12.1% in average, when compared to single-pass HD approaches, and has 13x fewer iterations on average than iterative HD approaches.

In the scope of this paper, conversely to previous works, we propose a new approach called "multi-centroid". More precisely, if the current data vector is more similar to an incorrect class than to any of the correct sub-classes, we create a new sub-class of the correct class. In this way, less common data patterns will have their own sub-class and will not get over-voted and

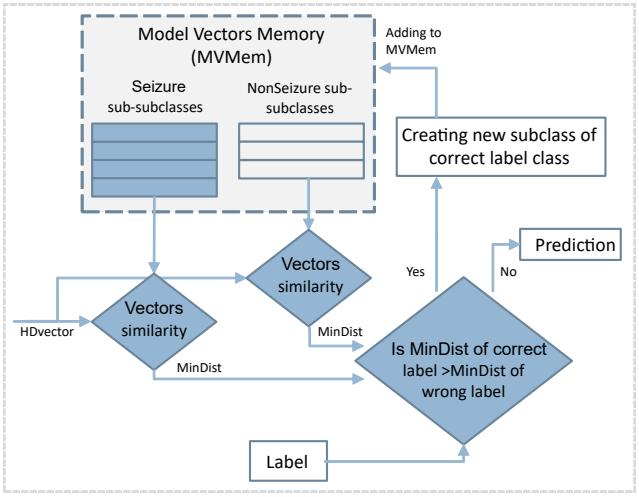

Figure 4.1: Schematic of the first step of multi-centroid training workflow where new sub-classes are created.

under-represented by more common patterns. This semi-supervised approach is guided by labeled data, but allows the creation of an unlimited number of sub-classes for each of the main classes. The number of sub-classes is highly dependent on the subject, data training instances, and also the amount of training data, and as such it would be hard to predict and set at the beginning as in [348].

Our proposed approach has a similar underlying idea as 'OnlineHD' [349], or iterative learning [227] in that it focuses on less common patterns. However, it is different since it allows the creation of sub-classes rather than adding them multiple times to a single vector. Consequently, our approach enables more control over the classification and potential interpretability of the predictions.

Proposed approach

The classical single-pass training procedure, as explained in Sec. 1.3.2 , for epileptic seizure detection, leads in the end to two prototype HD vectors: one for the ictal and one for the interictal class. This is most commonly used approach, but it has the drawback that all the training samples are equally important and summed up to the same class prototype vector. This leads to a dominance by the most common patterns of the prototype vectors, while less common patterns are under-represented. Thus, in our proposed multi-centroid approach, as illustrated in Fig. 4.1, we detect the difference of the current pattern/vector from the existing prototype vector. Then, in case of a significant difference, we create a new sub-class with its associated prototype vector. This significant difference is estimated by comparing the current vector with the prototype vectors of all the sub-classes of the correct class and of the wrong classes. If the most similar prototype is from a wrong class, then a new sub-class is created for the correct class, initialized with the current vector. As a result, our approach is also single-pass, thus the training procedure has the same complexity order as the classical one.

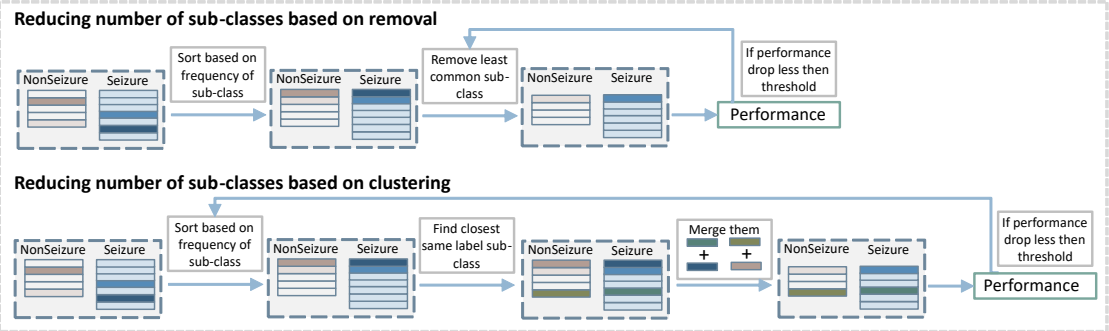

Figure 4.2: Diagram of the second step of our multi-centroid training workflow where number of sub-classes is reduced.

The first part of our approach creates new sub-classes without any additional constraint, except that new sub-classes have to be significantly different from the existing ones. As Fig. 4.3 shows, this procedure can sometimes result in a number of sub-classes that were created by just a few samples, and which might not contribute to the final prediction performance significantly. At the same time, they increase the memory requirements of the system as memory is linearly related to the number of HD prototype vectors needed to store. Thus, the second part of the algorithm removes some of the sub-classes while still keeping the increased performance benefits. Two methods were tested, as illustrated in Fig. 4.2: the first one is based on removing less common sub-classes, and the second is based on clustering them.

The approach to remove less common sub-classes starts by sorting classes based on the amount of data used to create them during training. Then, it removes them in steps, starting from least populated, while monitoring the performance after each step of removal. The performance is evaluated on the training set, and the iterative removal process is stopped once the performance drops for more than a pre-selected threshold.

Instead of removing less common sub-classes, the second approach merges them with the closest same label sub-class. The process is also performed iteratively. It starts from less common sub-classes, while monitoring the prediction performance and stopping the process in case of a performance drop bigger than a preset threshold.

4.2.3 Experimental Setup

Dataset used

As mentioned previously, our proposed multi-centroid approach is compared with the standard 2-class HD approach using EEG data on the use-case of epileptic seizures detection. We use the publicly available CHB-MIT database (as described in Sec. 2.1.2) to prepare three different datasets. Namely, often HD algorithms are tested on balanced versions of the databases where a sample of non-seizure data is randomly selected from raw data and matched in duration with seizure data. This often simplifies computation and performance assessment while

also allowing us to focus on the separability of the classes and preventing problems related to the class data distribution. Unfortunately, this does not represent real-life data distribution and can lead to a highly overestimated performance, which cannot be achieved during continuous monitoring with a wearable device. Thus, in this work, we use three different distributions of data: 1) balanced with an equal amount of ictal and interictal data, and 2) and 3) unbalanced with 5 and 10 times more interictal data. We call these three distributions F1, F5, and F10, respectively. During the balancing step of data preparation, when randomly selecting the interictal segments of data, we take care of not including data within 1 minute of seizure onset and up to 15 minutes after a seizure, as this data might contain ictal patterns.

Feature extraction and mapping to HD vectors

In standard ML approaches, more features usually lead to performance improvements [241], and this has also been shown for HD [350]. Thus, we use the same approach using 46 features as in [350] but with an additional feature of mean amplitude value. The initial 45 features, based on [241], contain 37 different entropy features, including sample, permutation, Renyi, Shannon, and Tsallis entropies, as well as 8 features from the frequency domain. For frequency-domain features, we compute the power spectral density and extract the relative power in the five common brain wave frequency bands; delta: [0.5-4] Hz, theta: [4-8] Hz, alpha: [8-12] Hz, beta: [12-30] Hz, gamma: [30-45] Hz, and a low-frequency component ([0-0.5] Hz), for each signal window. These features are commonly held to be medically relevant for detecting seizures [317].

Data is discretized into windows of duration W_{len}, for which features are calculated and encoded into an HD vector representing that data instance. This is repeated every W_{step}, i.e., a prediction is given based on W_{len} of data every W_{step}. In our case, W_{len} has been 8s, and W_{step} 1s.

Before HD training, vector memory maps need to be initialized. This means that we assign a static vector to each possible feature value, and to each combination of feature and channel. Features are normalized and discretized in the same number of levels, which allows us to use the same vectors to represent the values of all features. Thus, both feature value vectors HDV_{Val}, and feature-channel index vectors HDV_{ChFeat} representing a feature of a specific channel, are generated once before the training starts. More precisely, if we have M features and N channels, NxM HDV_{ChFeat} vectors will be initialized. HDV_{ChFeat} vectors representing features and channels are independently and randomly generated as there is no specific relation between features and channels. On the other hand, HDV_{Val} vectors are initialized in a way where first the vector is randomly initialized. Still, every subsequent vector representing the next possible value is created from the previous one by permuting consecutive blocks of d bits. The number of bits d depends on the number of possible needed values (and corresponding HDV_{Val} vectors). This approach ensures that vectors representing numbers that have closer values are also more similar.

Encoding is performed using ID-Level encoding as described with Eq. 1.1 in Section 1.3.2. More specifically, for each feature, its value vector HDV_{Val} and its index vector defining feature and a channel HDV_{ChFeat} are bound using XOR function. Finally, to get a final HD vector representing each data window, we bundle (sum and round) vectors of all features and channels, as shown in Fig. 1.13. In this approach, we do not distinguish between channels and treat them all equally important.

Validation strategy

Due to the subject-specific nature of epileptic seizures and their dynamics, the performance is evaluated on a personalized level. Data for each subject was pre-processed and divided into files, where each file contains one seizure, but the specific amount of non-seizure samples depends on the balancing type (1x, 5x, or 10x). This setting supports a leave-one-seizure-out approach, where the HD model is trained on all but one seizure/file. For example, for a subject with N_{seiz} files (each containing one seizure), we perform N_{seiz} leave-one-out training/test cycles and measure the final performance for that subject as the average of all cross-validation iterations.

Besides measuring the performance of seizure predictions, in this experimental analysis we also consider the number of sub-classes created and kept after the optimization steps of sub-classes removal or clustering, as well as the amount of data in them.

Performance Evaluation

The system's performance is quantified using several different measures to capture as much information as possible about predictions. Similarly as proposed in [244], [245] and later used in [350], we measure performance on two levels: 1) episode level, and 2) seizure duration level. Seizure duration is based on standard performance measures, where every sample is equally important and treated independently. The episode metric focuses, on the other side, on correctly detecting seizure episodes, but is less concerned about duration and correct prediction of each sample within seizure. This is, in detail, described in Section 2.3.5 and illustrated in Fig. 2.6.

As explained, for both levels, we measure sensitivity (true positive rate or TPR), precision (positive predictive value or PPV), and F1 score. Metrics on these two levels give us a better insight into the operation of the proposed algorithms. Furthermore, the performance measure often depends on the intended application and plays a big role in the acceptance of the proposed technology. Finally, in order to have a single measure for easier comparison of methods, we calculate the geometric mean value of F1 score for episodes ($F1E$) and duration($F1D$) as $F1DEgmean = sqrt(F1D * F1E)$.

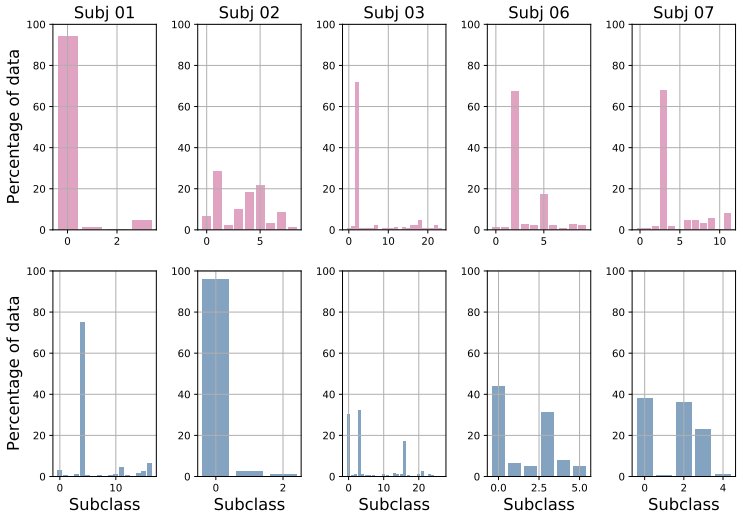

Figure 4.3: Percentage of data added to each of sub-class, shown for both seizure (red) and non-seizure (blue) sub-classes for 5 randomly selected subjects.

Label post-processing

We report the performance measures described in the above section for raw predictions. However, assuming that the decision of the classifier can change every second is not realistic from a real-time monitoring perspective. Thus, it is advisable to post-process labels before reporting performance figures. This is due to the viable time properties of the seizures, and also due to the small W_{len} and even smaller W_{step}, so that granularity is much smaller than the dynamics of the seizures. For example, it is not reasonable for seizure episodes to last only a few data samples, or if two seizures are very close, they probably belong to the same seizure and are so labeled by neurologists. Thus, as data and predictions are time sequences, we exploit time information to smooth the predictions by going through the predicted labels with a moving average window of a certain size SW_{len} (5s) and performing majority voting.

4.2.4 Results

Prediction performance

In Fig. 4.4, the performance between 2-class (2C) and multi-centroid (MC) models for all three data balancing cases (F1, F5, F10) is shown. Performance is reported as sensitivity, predictivity, and F1 score for both episode detection and seizure duration detection to get a deeper insight into the performance.

It is evident that the detection of episodes in the aspect of sensitivity is extremely high even for a 2-class model, so there is no real space for improvement with the multi-class approach. However, predictivity of both episodes and duration of seizures increases with multi-centroid model, meaning that fewer false positives are detected with MC approach than 2C approach. Only for duration sensitivity, even though there is an increase for the training set (not shown here), on the test set, we notice a slight decrease. Therefore, not the whole seizure duration

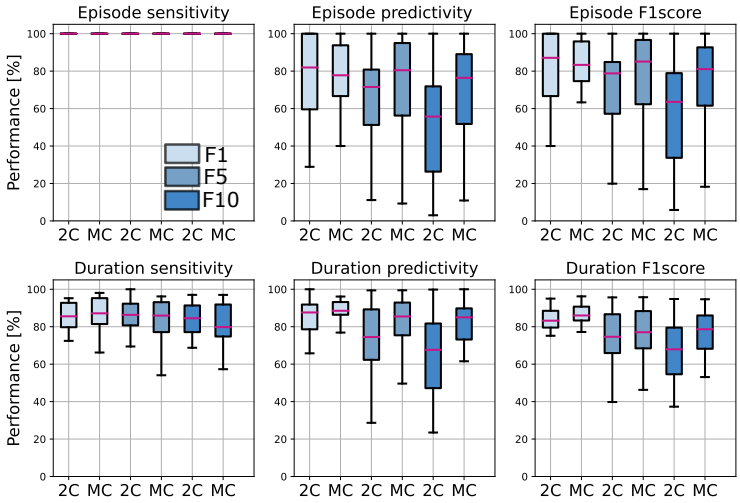

Figure 4.4: Average performance of all subjects in the test set, for 2 class (2C) and multi-centroid (MC) model. Performance measures shown: sensitivity, predictivity, and F_1 score both for episodes, and duration level.

is correctly predicted. Further, it can be noticed that performance is, in general, worse for non-balanced datasets and that performance drops with more non-seizure data. Therefore, it is required to report all three performance values, as reporting only performance on the balanced dataset (as most works in the literature do) can lead to misleading results about the performance on real-life data distribution. Moreover, the performance increment due to the multi-centroid approach is higher for more unbalanced datasets (F10 and F5 when compared to F1), which can be explained by the initially higher space for improvement.

Analysis of created sub-classes

Fig. 4.3 shows the number and distribution of sub-classes created during multi-centroid training, for both seizure and non-seizure sub-classes for few subjects. First, we can see that the number of sub-classes created is very variable among subjects, some having only a few (e.g., Subj 1 with 4 sub-classes for seizure) and some having a lot of them (e.g., Subj 3 with more than 20 seizure sub-classes). The number of seizure and non-seizure sub-classes is also very variable within the subject. For example, subject 1 has 17 non-seizure sub-classes and only four seizure sub-classes, while subject 7 has 11 seizure sub-classes and only five non-seizure ones. This situation reflects the variability of raw data and demonstrates the rationale for our multi-centroid approach instead of grouping all seizures into one vector (class) and all non-seizures to another vector, as done in the 2-class model.

Furthermore, it is very interesting to observe the amount of data used to create each of the sub-classes. This corresponds to the frequency of occurrence of each sub-class and shows that there are usually 1 to 3 sub-classes that are very common, while the rest are less common. This also varies greatly between patients, and whether it is a seizure or non-seizure class. This is the main motivation behind the strategies we implemented to reduce the number of sub-classes.

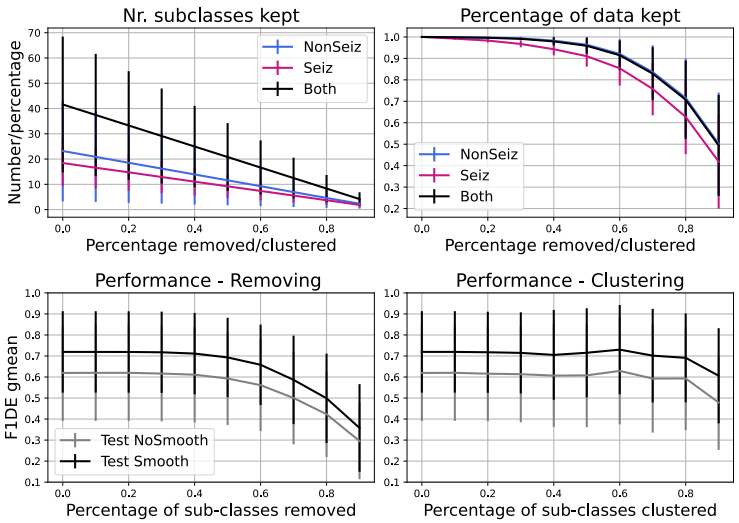

Figure 4.5: Iterative reduction of sub-classes and how it affects the number of sub-classes and percentage of data in them. Further, performance decrease through steps due to removing/clustering of sub-classes is shown. Performance for test set before (gray) and after (black) smoothing is shown.

Reduction of sub-classes

In Fig. 4.5, results of the experiment are presented, where we iteratively remove 10% of less common sub-classes in every step of the iteration. We see that the number of seizure and non-seizure sub-classes is linearly dropping, while the percentage of data retained is slowly dropping at the beginning and faster later. More specifically, it is reduced more quickly for seizure data as sub-classes are more evenly populated than non-seizure sub-classes. In total, significant data reduction begins to occur once >50% of sub-classes are removed.

For the same experiment, we show how the performance (F1DEgmean) decreases while iteratively removing or clustering sub-classes. Similarly, there is no significant drop in performance up to 50% of sub-classes being removed, and after it drops very steeply. The decrease is less steep for clustering and allows even 80% of sub-classes to be clustered while keeping high performance (meaning the drop that is smaller than 5% for gmean of F1 score for episodes and duration).

Optimizing performance and number of sub-classes

As shown in Fig. 4.5, it is possible to reduce the number of sub-classes significantly, while not sacrificing much in terms of performance. Thus, as explained in Sec. 4.2.2, we tested two approaches. The first approach (MCr) removes the less common sub-classes iteratively in steps (10% of sub-classes in each step) and, after each iteration, evaluates performance on both training and test set. If the performance on the training set drops more than a given tolerance threshold (in this case, 3% of F1DEgmean was used), the process is stopped and the number of sub-classes is considered optimal. The second approach (MCc) is based on clustering the less common sub-classes rather than completely removing them. In this approach, as well in iterative steps, we pick the less common sub-classes (10% of them in each step) and merge them with the most similar sub-classes of the same global label to each of them. The

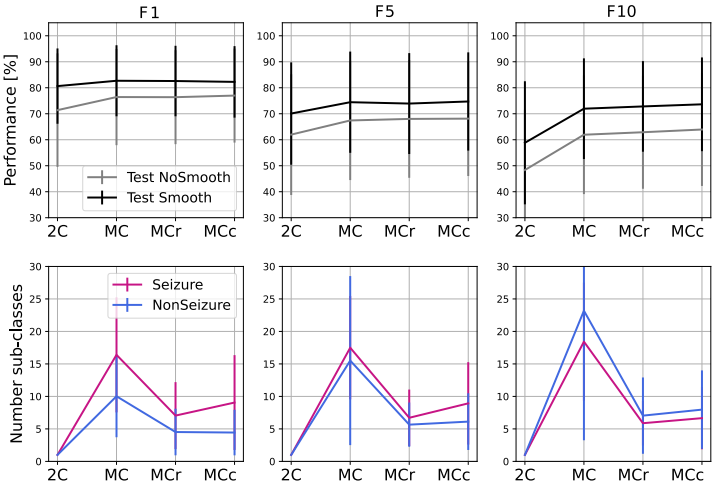

Figure 4.6: Performance and number of sub-classes for 2-class (2C), multi-class (MC) approach and with 2 methods for reduction of number of sub-classes: sub-classes removal (MCr) and clustering (MCc).

process stops after performance drops more than a tolerance threshold in the training set, the same as in the MCred approach. In Fig. 4.6 we show performances and number of sub-classes for 2-class model (2C), initial multi-centroid (MC) model, and after two approaches for optimization, with sub-classes reduction (MCr) and clustering (MCc). Only one performance is shown, F1DEgmean, to simplify comparisons. Finally, we report the results for the test set using all three balancing scenarios (F1, F5, F10).

Based on the results in Fig. 4.6, we can conclude that the approaches to optimize the number of sub-classes do not significantly degrade performance when compared to the multi-centroid model, which is still substantially higher than the 2-class model (for F5 and F10). P values of Wilcoxon statistical test for difference between 2C and MC approach (after smoothing) were 0.159, 0.009 and 1.19e-16 for F1, F5 and F10 data balancing, respectively. On the other hand, the number of sub-classes is much smaller when compared to the multi-centroid model. Even though the iterative process and number of sub-classes were decided based on training data performance, test data performance also remains equivalent with the MC model. The number of sub-classes is reduced by 50% (or more) in all three dataset balancing cases. The sub-class removal approach leads to a slightly fewer sub-classes than the clustering approach.

In Fig. 4.7, we summarize our experimental results and show the final performance improvement and the number of sub-classes after the multi-centroid model with the removal of sub-classes (MCred) for all three balancing datasets. Performance improvement is the smallest for F1, as the space for improvement is also the smallest. For highly unbalanced data (F10), an increase of up to 14% (on top of initial performance) for the test set was achieved. The multi-centroid approach has the biggest potential for more unbalanced datasets, which are closer to real-life data distribution and also have the lowest absolute performance. As seen in Fig. 4.6, F1 performance is improved from 80% to 83% for the test set after smoothing, and from 60% to 73% for F10.

When observing the number of sub-classes for different dataset balancing strategies, it seems

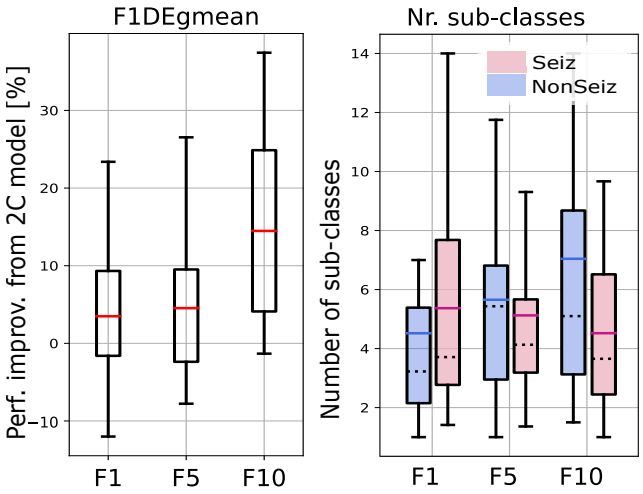

Figure 4.7: Summarized results for multi-class learning when compared to 2-class learning. Performance improvement and number of sub-classes is reported for all three balancing datasets.

that the more non-seizure data considered, the more sub-classes are necessary. This conclusion is logical, as we add more data that can represent different neural activity states. On the other hand, in terms of seizure sub-classes, the more non-seizure data we have, the fewer seizure sub-classes we need, as training is less sensitive to small changes in seizure dynamics. Thus, our results indicate that the multi-centroid approach is more significant the closer we are to more realistic data balancing.

4.2.5 Conclusion

In this chapter, we have presented a novel semi-supervised learning approach aimed at improving hyperdimensional computing models. The multi-centroid approach was tested on the challenging use case of epileptic seizures detection. In particular, based on given global labels (seizure or non-seizure), instead of forcing only two HD prototype vectors, one for each class, we allow unsupervised creation of any number of sub-classes and their centroid vectors (of seizure and non-seizure). This enables less common signal patterns not to be under-represented, but to create their own sub-class when they are significantly different from the existing sub-classes.

Our proposed multi-centroid approach has significantly improved performance when compared to a simple 2-class HD model; up to 14% on the test set of the most challenging dataset with 10 times more non-seizure than seizure data. It also leads to the creation of a highly variable number of seizure and non-seizure sub-classes for each subject, reflecting the complexity of the data and the classification challenge itself. One drawback of this approach is the memory requirements that storing all sub-class model vectors implies. However, this increment is linear with the number of sub-classes and can be easily constrained according to the hardware requirements of different types of possible final wearable platforms.

Then, we designed and tested two approaches for optimizing the number of sub-classes, while still keeping an improved performance, as well as sub-classes reduction and clustering. Both

approaches have led to a significant reduction of the number of sub-classes (~50%), while maintaining equally high performance as the first expanding step of the initial multi-centroid model.

Finally, the multi-centroid has proven to be able to reach bigger improvement for less-balanced datasets. At the same time, the total number of sub-classes is not significantly increased compared to the balanced dataset. Thus, it can be an important step forward to achieve high performance in epilepsy detection with real-life data distributions, where seizures are infrequent, especially during online learning.

4.3 Comparison of HD computing learning strategies

4.3.1 Motivation

As we have seen in previous chapter 4.2 standard HD computing approach is not performing so well when taking into account more realistic data distributions [196].

At the same time, various strategies have been proposed in the literature to improve learning and detection with HD computing. For example, the iterative learning approach [227] has been proposed instead of single-pass learning, but it has not been fully explored yet for epilepsy. The 'OnlineHD' approach [349] also works in a single-pass manner but multiplies vectors with a weight factor depending on the novelty each data window brings. It aims, as well as iterative learning, to overcome the dominance problem of more common patterns in the final prototype vectors, but has also not been tested on the epilepsy detection task. In the end, we want to compare those approaches with the above proposed multi-centroid approach.

Thus, following the challenges epileptic seizure detection poses, we aim to test and compare various algorithms for HD computing learning improvement, hoping to achieve a performance that would be acceptable to users. In this work, we contribute to the state of the art in the following manner:

- We systematically compare the performance of the standard single-pass one-centroid HD computing approach with the classical random forest learning model on epileptic seizure detection with more realistic data distributions and show that HD computing still has a performance gap to cross.
- We then implement several existing proposals for improved HD computing learning: iterative (multi-pass) one-centroid learning, 'OnlineHD' as well as multi-centroid approach. Not all of them have been yet tested for epileptic seizure detection.
- We combine two strategies to test whether multi-centroid with multi-pass approach can bring additional improvements to the performance.
- Finally, we compare the mentioned strategies in terms of their performance and their memory and computational requirements with a wearable implementation in mind.

4.3.2 HD computing learning strategies

Traditionally, HD computing classifiers have been based on single-pass learning and a single-centroid model vector per class, as illustrated in Fig. 4.8A. However, many current machine learning problems are challenging due to the immense complexity and variability of patterns in data, especially when compared to the amount of training data available. Epilepsy is one such case, where electroencephalogram (EEG) signatures of epileptic seizures are highly unique and variable among people, brain states, and time instances, especially if they are grouped under only two given labels. Non-seizure data represents many different brain states, such as awake, sleeping, physical or mental effort conditions, which can significantly increase

data variability grouped under one class. All of these states have their own brain signatures that need to be learned. In the standard single-pass one-centroid HD approach, all data samples are equally important during learning, leading to more common patterns dominating the prototype vectors. This means that less common patterns could be potentially under-represented and wrongly predicted even on the same training data. Thus, in this work, we compare different proposed strategies that can tackle this problem: iterative (multi-pass) learning, multi-centroid, multi-centroid with iterative learning, and 'OnlineHD'. We find epileptic seizure detection a perfect test case for this due to the variability of patterns in both seizure and non-seizure classes, the inherent disbalance in the amount of seizure/non-seizure recordings, and a generally relatively small amount of seizure training data.

Multi-pass learning approach

The first approach aimed at overcoming the problem of dominance of common patterns is iterative learning [227]. We call it also multi-pass learning to clarify its relation to other approaches. In the first pass of learning, all samples are added to the corresponding data class, the same as in traditional single-pass learning. Next, an iterative process of multiple passes starts wherein each pass for each sample prediction is given based on the learned prototype vectors from the previous pass. In case of a wrong prediction, this sample is added again to the correct class. In this way, less common patterns that got under-represented are strengthened by adding them multiple times. After each pass, performance is evaluated on the same training set. The multi-pass process stops once there is no more significant improvement in performance. The workflow of iterative learning is shown in Fig. 4.8B.

This approach can be improved if the vector of a wrongly classified sample is added to the correct class prototype vector and also subtracted from the other ones. Further, before adding/-subtracting vectors to prototype vectors, it can be multiplied with a factor (learning rate) to increase its weight. In [227], authors test different values of multiplication factors (learning rate) and their influence on the number of iterations needed to achieve a stable performance and the final performance itself. They show that small factors lead to clear performance increase but require many iterations. On the other side, too big multiplication factors lead to fluctuations in performance, thus potentially never converging and never finishing the training procedure. Hence, they also tested an adaptable threshold strategy to exploit the advantages of a large learning rate at the beginning and a small learning rate at the end for fine-tuning the performance. They tested these approaches on four different classification applications and showed performance improvement as well as improved energy and computational complexity. Unfortunately, none of the applications used for testing was epileptic seizure detection, and one could argue that the chosen problems were less complex and challenging than the epileptic seizure one. Further, in order to be able to compare properly with other approaches (explained below), we implemented this approach as well.

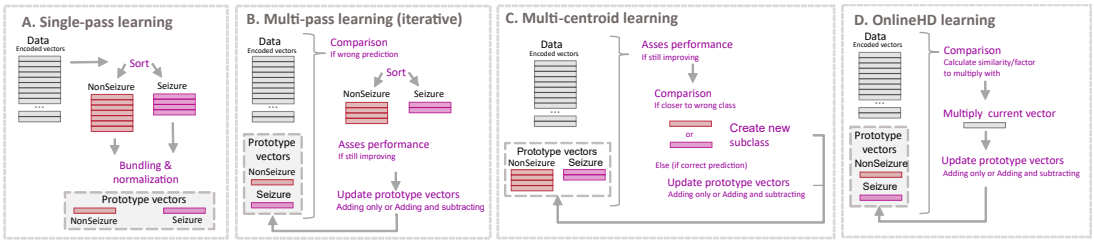

Figure 4.8: Schematic of different improvements on HD computing learning: A. Standard single-pass single-centroid HD learning, B. Multi-pass (iterative) single-centroid learning, C. Multi-centroid single-pass learning and D. 'OnlineHD' single-pass learning.

Multi-centroid learning approach

Another way to not let the frequency of certain patterns make them dominate the model, leading to under-representation and worse classification of less common patterns, is to treat them as separate sub-classes (centroids) of the same class. In this way, they are not all accumulated to the same prototype vectors, but it is possible to have multiple vectors representing the same class. This approach was explored in the previous section Sec. 4.2 and has been shown to significantly improve epileptic seizure detection. Balanced datasets, with the same amount of seizure and non-seizure data, are commonly used, but are not realistic in real-life applications. Thus, we compared performance also on unbalanced datasets with more non-seizure than seizure data. The more unbalanced the dataset was, the bigger the performance improvement when compared to the single-centroid approach.

The workflow of the approach is illustrated in Fig. 4.8C. This approach is interesting as it allows the semi-supervised creation of an unlimited number of centroids (sub-classes) of each class. Namely, if the current sample vector is more similar to the wrong class than any correct sub-class (meaning that it would be wrongly classified), a new subclass of the correct class is created. This process can sometimes lead to many sub-classes where some have only a few samples that contributed (were added) to it. Hence, in the last step of the workflow, the number of sub-classes is reduced by either removing the less common sub-classes or by clustering them with the most similar sub-classes sharing the same label. This step significantly reduces the number of sub-classes and thus the memory required to store them while still achieving higher performance by allowing several centroids. When compared to the iterative multi-pass approach, this approach is more memory-consuming but should be faster due to the single-pass approach. This is the motivation to compare those two approaches in more detail.

Furthermore, it is interesting to test whether combining iterative and multi-centroid approaches could further improve performance. Namely, the multi-centroid and multi-pass approach would consist of the first step with one pass of multi-centroid training and then passing several times through the training data to fine-tune the exact centroid values.

Weighted learning approach

Following the data-driven learning rate idea from [227], in [349], the multi-pass approach is replaced with single-pass to reduce the training costs. More specifically, as shown in Fig. 4.8D., a naive accumulation of equally important samples is replaced by using the weighting approach before adding the current vector to the prototype vectors. The weight is defined by the similarity of the current vector to the current prototype vectors; the higher the similarity, the lower the weight. This approach identifies the most dominating patterns and lowers model saturation by them. In [349], authors compared this so called 'OnlineHD' approach in performance to the previously described iterative approach. Comparable accuracy was achieved on several different classification problems. This approach should be as memory-consuming as the traditional one-centroid single-pass, but might be more time-consuming. This is due to the need to update continuously and normalize the prototype vector after adding each training sample (and not at the end of the pass as in other approaches). It is interesting to analyze the performance of this approach and compare it to the two previously proposed methods for preventing the under-representation of less common patterns.

4.3.3 Experimental Setup

Database and features

We utilize CHB-MIT database, as described in Sec. 2.1.2. From the raw database, we prepared a dataset that contains ten times more non-seizure data than seizure data to be closer to a more real-life data balance. We skipped a balanced scenario,as it can lead to a highly overestimated performance, not achievable during the continuous monitoring with a wearable device [196]. Non-seizure segments were chosen randomly from available non-seizure data, but excluding data 1 min before and 15 min after a seizure. This data might contain ictal patterns and thus make classes less separable.

In this chapter, we use 46 features as used in the previous chapter to make it comparable. The encoding of features and their values to HD vectors is the same as in the previous chapter as well.

Validation

Training and evaluation are performed in a personalized manner. This is due to the subject-specific nature of epileptic seizures and their signal patterns. More specifically, for each subject and seizure, data is preprocessed to contain 10 times more randomly selected non-seizure data and saved to an individual file. Then leave-one-seizure-out cross-validation is performed, where HD models are trained on all but one file (containing one seizure each). Final performances reported per subject are the average of all cross-validation iterations. For the performance measures, we use the same measures as explained in the previous chapter as well as in Sec. 2.3.5. Identical label postprocessing is employed too.

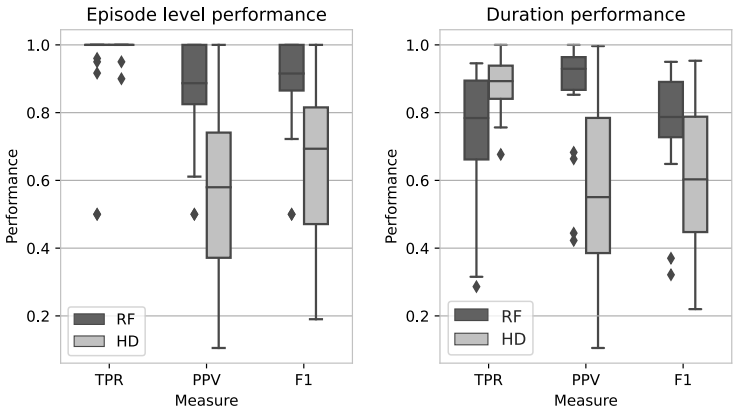

Figure 4.9: Performance of random forest as a benchmark for HD computing. Performance is shown for both episode and duration level, including sensitivity, precision and F1 score values.

Statistical analysis

Due to the high variability in performance between subjects, we perform statistical analyzes to compare different strategies. We compare each learning approach with the traditional single-pass one-centroid approach using the Wilcoxon statistical test. It compares the performance of two paired groups, and we report the p-value.

4.3.4 Results

Standard HD computing learning

Fig. 4.9 shows a performance comparison of the standard single-pass single-centroid HD computing approach with Random Forest performance as a benchmark. The random forest contained up to 100 trees and was trained on the same dataset, with identical preparation, train-test split, and post-processing as the HD computing approach. Here we see that HD computing is performing significantly worse than the random forest approach. TPR or sensitivity stays equally good with the HD approach, meaning that all seizures are usually detected, but a significant drop is perceived in PPV (precision), which means that many false positives occur when using HD computing. For F1 score for episodes the drop in the mean performance of all subjects is 25.7%, while for F1 duration score is 16.9%. Thus, in the next experiments, we test if improved learning strategies of HD learning can help resolve this problem and reach the performance of random forest.

Multi-pass (iterative) learning

In Fig. 4.10, two versions of the multi-pass approach are compared with the standard single-pass approach. In one approach, when the current data window would be wrongly classified, it was just added to the correct class again ($2C^{+}$), whereas, in another approach, it was also subtracted from the wrong class ($2C^{+}_{-}$). The average number of iterations, percentage of re-added data, and prediction performance are analyzed and shown as a distribution over

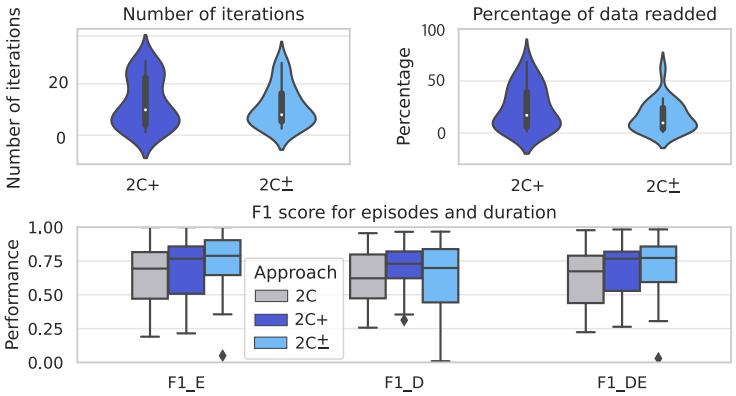

Figure 4.10: Iterative (multi-pass) approach compared to single-pass approach. The average number of iterations, percentage of data that had to be re-added, and performances on the test set are shown.

all subjects. It can be observed that on average, it took around 13 passes for $2C^+$ and 12 passes for $2C^+_-$ to get a stable performance value. The amount of data that had to be re-added on average was 24% for $2C^+$ and 16% for $2C^+_-$. Interestingly, for $2C^+_-$ model, the variability between patients is smaller than for the $2C^+$ model. Concerning the performance, the F1 score for episodes ($F1_E$) and for duration ($F1_D$) as well as their mean ($F1_{DE}$) is shown. Both multi-pass ($2C^+$ and $2C^+_-$) approaches have significantly higher performance when compared to single-pass ($2C$) approach ($p = 1.63e^{-5}$ for $2C^+$ and $p = 2.6e^{-2}$ for $2C^+_-$ for mean of $F1_E$ and $F1_D$). There is no significant difference in $F1_{DE}$ performance between between using either the $2C^+$ or $2C^+_-$ approach ($p = 0.74$).

Multi-centroid learning

Fig. 4.11 shows the performance of several versions of a multi-centroid approach when compared to a traditional single-centroid one. As described in chapter 4.2 first step of the multi-centroid approach is to allow an unlimited creation of sub-classes (MC), after which the essential step of removing unnecessary sub-classes follows. This can be done either by simply removing the least common centroids of each class (MCr) or clustering (MCc) least common centroids with the closest same-class ones. Here, results for only MCr approach are shown as they were slightly better and it is lighter to implement.

Results show that the first step of MC approach creates, in average for all subjects, 18 sub-classes for seizure and 22 for non-seizure. The approach of removing least populated classes (MCr) results in, on average, five seizure and six non-seizure sub-classes. Variability is very high between subjects and is within 6 and 117 sub-classes (summed for seizure and non-seizure) after the first MC step, and is reduced to range from 2 to 26 after MCr step. Going from single-centroid to multi-centroid significantly improves the performance on both train and test sets. More specifically, for the test set the average performance of all subjects is increased from 61% to 76% for mean value of F1 score for duration and episodes. It is important to note that after the second step of reducing the number of sub-classes, despite a significant reduction in the number of sub-classes, the performance is not significantly reduced for the MCr approach.

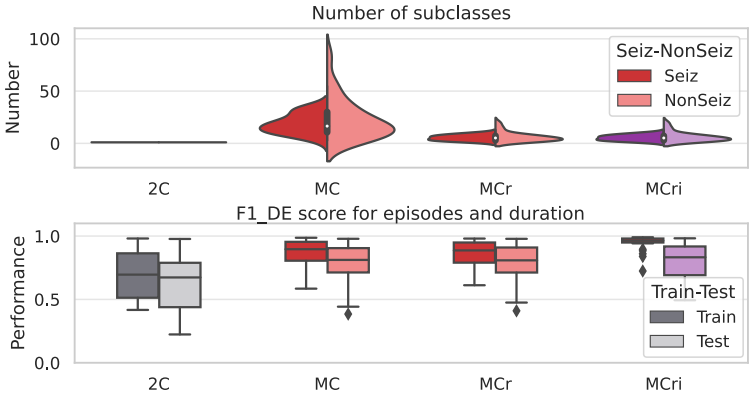

Figure 4.11: Multi-centroid learning approach is compared to traditional single-centroid approach. The average number of centroids per seizure and non-seizure as well as performance is shown.

Next, we tested whether performing additional steps of iterative learning on centroids decided after MCr can lead to additional performance improvements. Thus, the number of centroids (and their initial structure) was fixed based on the previous step, but then by passing several times through the training dataset, we allowed slight fine-tuning of the centroids. Although additional performance improvement can be obtained in the training set, but not in the test set. We believe this is due to overfitting.

Weighted learning

The weighted approach, also called 'OnlineHD' [349], is an alternative to the multi-pass and multi-centroid approaches. In Fig. 4.12, the distribution of the weights for seizure and non-seizure is shown. The average values of weights are between 0.1 and 0.2, meaning that between 80% and 90% of the bits in vectors were identical when adding a new sample. For some subjects, such as subjects 5, 9, 10, 14, 19 and 24, values are similar for both seizure and non-seizure, while for some subjects, seizure values are larger than for non-seizure, meaning that they were bringing more novelty. For a few subjects (e.g., for subj 1 and 23), it was the opposite, i.e., non-seizure data brought more novelty.

Finally, when comparing performance, we compared two versions of weighted learning. The first one where the weighed sample is added only to the correct class (On^{+}) and the second one, similar to multi-pass, where it is also subtracted from the wrong class if it would be wrongly predicted (On^{+}_{-}). Performance results in Fig. 4.12 show that just weighted adding (On^{+}) does not significantly improve results on the test set, but that adding and subtracting does help significantly.

Comparison of all tested learning strategies

Finally, in Fig. 4.13, we compare the seizure detection performance for all the studied strategies. The mean of F1 score for duration and episodes is shown as a distribution over all subjects. Again, the performance of standard HD computing ($2C$) with single-pass and single-centroid leads to a significantly lower performance and higher inter-subject variability than the random

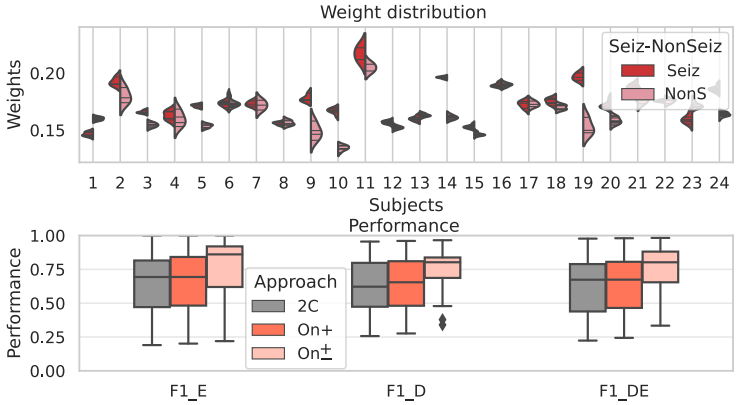

Figure 4.12: Weighted learning approach is compared to the traditional single-centroid approach. The distribution of the weights for seizure and non-seizure is shown for each subject on the upper plot. The lower plot shows the performance when comparing two weighted approaches with the traditional (non-weighted) approach.

forest (RF) approach. Performing iterative (multi-pass) learning improves the performance both for training and test sets, but a larger improvement is possible when both adding the current data vector to the correct class and subtracting it from the wrong class ($2C^{+}_{-}$) than only adding it ($2C^{+}$), as shown previously in the literature.

The multi-centroid approach (MC, MCr) also leads to a significantly improved performance (with respect to $2C$) but not yet achieving the level of random forest. If centroids are then fine-tuned through several passes of multi-pass learning ($MCri$), performance is improved even more. In that case, there is no significant difference between multi-centroid multi-pass approach with removal of less common classes ($MCri$) with respect to random forest ($p = 0.18$).

The 'OnlineHD' approach also leads to a significant improvement in performance, but only when using the approach of adding and subtracting vectors from correct and wrong classes, respectively. However, it did not reach the level of no significant difference with random forest ($p = 0.013$).

These results show that with additional improvements to standard HD computing, it is possible to achieve a performance as good as with random forests for the seizure detection problem. Nevertheless, Fig. 4.13 highlights that variance between subjects is high, especially for HD, and there is space for further improvements of HD performance.

Memory and computational requirements

Finally, we analyse also the memory and computational complexity of the different learning strategies. Memory is calculated as the memory needed to store all the prototype vectors. In Fig. 4.14, the relative amount of memory is shown to compare different approaches. Multi-centroid learning (MC) obviously requires the largest amount of memory, but this is reduced to less than half after optimization steps of removing unnecessary centroids in the second step of the algorithm (MCr). However, this is still five times (on average) larger memory overhead in comparison to the single-centroid 'OnlineHD' ($On+$ or On^{+}_{-}) approach.

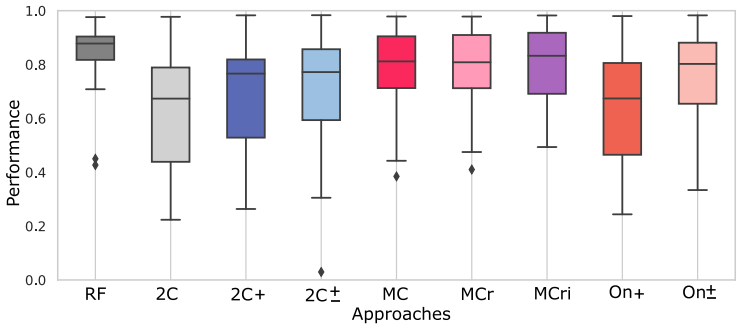

Figure 4.13: All learning approaches tested in this work are compared together regarding seizure detection performance. Only one measure combining all together is shown (mean of F1 score for episodes and duration). Results on the test set are shown, where boxplots represent a distribution of performance over all subjects.

One thing to notice when thinking about reducing memory for wearable applications is that after the training is done, it can be analyzed which particular bits of HD vectors are different between the two classes. Only those bits contribute to prediction. Thus, in practice, if HD models will not be updated anymore (as in the case of online learning), the dimension of HD vectors can be reduced in the testing part. This would mean that the dimensions of all vectors (initial vectors used to create data representation, data vectors, as well as class prototype vectors) could be reduced by removing unnecessary bits, further reducing memory requirements.

In order to quantify computational complexity we analysed the time needed to perform each type of training using the Python implementation running on a single-core. We evaluate the relative time between learning strategies. Time was measured as an average over all subjects and all cross-validation iterations. As expected, iterative, multi-pass learning ($2C^+$ or $2C^+_-$) is significantly more complex (>30x time for $2C$), due to the need for evaluation of performance after each pass of training. Multi-centroid approach in its first step (MC) is not extremely time consuming (~5x time for $2C$), but the second step (Mrc) of optimizing the number of centroids increases the computational complexity (>35x time for $2C$). In the end, multi-centroid and multi-pass ($MCri$) is understandably the most time and computationally intensive (~85x time for $2C$). Interestingly, 'OnlineHD' (On^+ or On^+_-) learning, even though more complex than single-pass, is significantly more lightweight than the multi-pass or multi-centroid approaches (only ~2x time for $2C$).

4.3.5 Conclusion

In this work, we have investigated and compared the characteristics of different HD computing strategies in the context of epileptic seizure detection with respect to more established approaches such as a random forest. In particular, our results have shown a significantly lower performance of the standard single-pass single-centroid HD approach compared to the random forest one, mainly due to many false-positive seizure detections.

All three investigated ideas for improving learning capabilities of HD computing, namely multi-

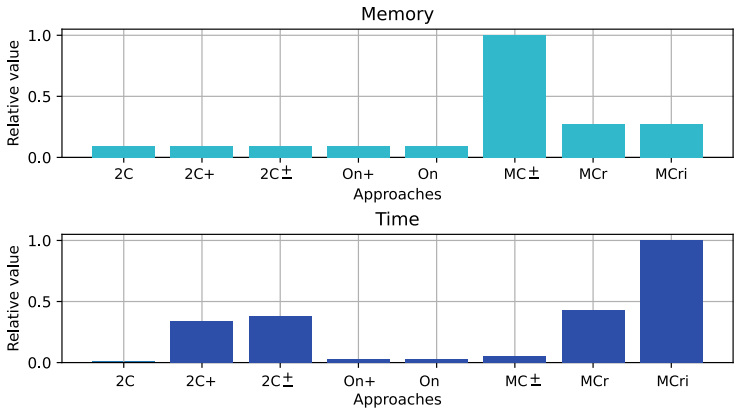

Figure 4.14: Comparison of different learning approaches with respect to the time needed for calculation of prototype vectors and memory required for storing them.

pass, 'OnlineHD', and multi-centroid, do indeed improve performance even in the unbalanced dataset. Not all reach the performance of RF, though. Experiments showed that both for multi-pass and 'OnlineHD' training, during misclassification of a sample, subtracting that sample vector from the wrongly predicted class, and not just adding it more times to the correct class, significantly improves the models. For multi-centroid, combining it with iterative multi-pass learning improves results even more so that performance is not significantly different from that of the random forest model.

Finally, the memory and computational complexity of the three approaches were compared. Multi-pass and multi-centroid are similarly computationally complex. Multi-centroid is also more memory-demanding than the other two approaches. Overall, the 'OnlineHD' approach is much more memory and computationally-friendly when compared to a multi-pass approach, which takes more time to train, or when compared to a multi-centroid approach, which requires more memory.

Nonetheless, these analyses and results have proven the potential HD computing has for real-life epileptic seizure detection.

4.4 HDTorch

4.4.1 Motivation

From an algorithmic perspective, many HDC variations have been explored in literature, touching on almost every aspect of the HD training or inference flow. These include methods for initializing hypervectors [187], [240], [350], accumulating datapoints into class vectors [186], [191], [192], [196], [233], and calculating similarity between data and class vectors [351]. Exploring this design space necessitates analysis of the effects of various hyperparameter values such as hypervector lengths or online batch sizes, testing different encoding and learning strategies, and applying pre- or post-processing filtering to data or results. Unfortunately, due to the lack of publicly available libraries for fast processing and parallelization of HD computing on CPU or GPU, such analysis has thus far been performed on unoptimized HDC frameworks, greatly reducing research efficiency.

For example, to the best of our knowledge, HD computing has not been tested on large datasets such as those for epilepsy detection. While previous works applied HD computing to epilepsy datasets, data subsets were always utilized [99], [204], [240], [269], [350]. Unfortunately, as has just been demonstrated in previous chapters, training on subsets of data may result in significant alterations to the final predictions in comparison to utilizing the entire dataset [196]. Thus, to develop suitable HDC algorithms, accelerating the exploration of the HDC design space is necessary.

Moreover, let us look at the typical HD worflow presented in Sec. 1.3.2. We will utilize as an example, subject 1 of the CHB-MIT epilepsy database consisting of 40 hours of data, split in 1-hour segments as in the original database. We use the HDC framework from [269], with ID-Level Encoding and Hamming distance similarity measurement. We perform both classical and 'OnlineHD' training using the common leave-one-out cross-validation strategy.

The results of CPU training and inference are as follows. Classical training takes 46 minutes, 22 of which are consumed by the encoding step. Online training takes 134 minutes, 28 of which are consumed by encoding. If extrapolated to the full 980 hours of data, with a time resolution of 0.5 seconds, resulting in more than 7 million samples included in the CHB-MIT database, classical/online training would take 19/54 hours, respectively.

It is no wonder, then, that previous works have analyzed only subsets of the dataset, as performing any sort of design space exploration is infeasible with such high runtimes. Thus, a GPU-accelerated, flexible HD computing framework is necessary to enable future fast, iterative research into the HDC design space.

To conclude, the highly parallel nature of HDC algorithms lends motivation to the development of specific HDC hardware accelerators. While such accelerators will certainly be implemented in future products that rely on HD computing, they are expensive and limited in algorithmic flexibility, a necessity for research into the HDC design space. Therefore, open-source, flexible

GPU-accelerated HDC frameworks are necessary to enable efficient HDC research.

In this context, we propose HDTorch, the (first) open-source, PyTorch-based library built for exploring the HDC paradigm. HDTorch unlocks the full potential of PyTorch applied to HDC algorithms, and further extends PyTorch with custom, CUDA-backed hypervector operations. HDTorch is highly customizable, enabling modification to hyperparameters and encoding/similarity strategies. We validate HDTorch's accuracy and runtime performance on four reference HDC benchmarks, demonstrating accuracy comparable to state-of-the-art works while greatly accelerating training and inference for classical and online HD strategies.

At the same time as we worked on HDTorch, another group motivated by the same lack of GPU libraries for HD algorithms developed TorchHD [315]. It is similar to HDTorch, a Pytorch-based library, but without custom CUDA extensions for memory reduction and additional speedup, as will be explained in the next sections. This demonstrates a clear interest in developing standardized tools for HD computing, which will likely converge towards a large community that will maintain them and makes them useful to a broader machine-learning public.

We further motivate HDTorch's utility for exploring the HDC design space by applying it to the CHB-MIT epilepsy database. HDTorch enables us to perform the first ever HDC training on the entire dataset by reducing training and inference time by over 70x. We draw several conclusions from the analysis that will be useful for future works applying HD computing to large, unbalanced datasets.

This work's contributions are summarized as follows:

- We introduce HDTorch, an open-source, PyTorch-based HDC framework with CUDA extensions for hypervector operations, namely, bit-(un)packing and bit-array summation in the horizontal/vertical dimensions.
- We benchmark classical/online HD computing with HDTorch, showing accelerations of (111x/68x)/87x for (classical/online) training and inference, respectively.
- We explore the accuracy/runtime impact of varying hyperparameters, namely, hypervector width and online HD training batch size.
- We motivate HDTorch's design space exploration utility by performing, to the best of our knowledge, the first HDC training on the entire 980-hour, 7 million datapoint CHB-MIT epilepsy detection dataset. We present novel observations on the utility of HD computing on large, unbalanced datasets.

4.4.2 Accelerating HD Computing

Several works have begun to explore strategies of accelerating HD computing via various software and hardware frameworks. Works mainly focus on 1) ASIC implementations [188], [197], [201], [352] or 2) in-memory computing accelerators [332], [353], [354]. Other works explore GPU acceleration of HD computing. In [192], the authors implement an HDC architecture in PyTorch to compare against other ML algorithms on an embedded GPU, analyzing runtime,

Table 4.1: HDTorch Feature Overview

Features	Values
Customizable Hyperparameters	Hypervector Dimension Batch Size
Hypervector Flavors	Binary (0,1) Bipolar (-1,1)
Hypervector Generation Strategies	Random Scale Random [187], [350] Sandwich [269]
Available Binding Strategies	ID-Level Encoding [330] Feature Permutation [233] Feature Appending [305]
Available Similarity Metrics	Hamming [351] Cosine [351]
HD Computing CUDA Extensions	Binary (Un)Packing Horizontal/Vertical Summation

memory usage, and energy consumption. Similarly, the authors in [186] design a TensorFlow framework with HDC-specific extensions to accelerate the encoding and training process.

Our proposal differs from the previous two in several ways. First, we utilize PyTorch as a base framework for HDTorch, and second, as we want to encourage fast design space exploration in future works, we rely on native PyTorch operations where possible, and develop a set of custom CUDA functions where we identify HDC-specific bottlenecks that PyTorch cannot handle effectively. These functions are different from those accelerated in previous papers.

Implementing HD Computing in PyTorch

PyTorch is one of the two most popular Deep Learning (DL) frameworks utilized in research today, along with TensorFlow. While the debate on framework superiority is contentious, there can be no doubt that PyTorch is currently the most popular DL framework in research, being utilized in over 75% of research papers using either of the two frameworks [355]. As such, providing HDC support in PyTorch will enable the majority of the research community to explore HDC solutions on a wider range of topics more easily.

HDTorch is realized as a python library installable from PyPi[I]. Table 4.1 lists the features provided by HDTorch's base HD model class. Namely, it supports variable size binary or bipolar hyperdimensional vectors for any number of classes. Value and feature hypervectors may be generated randomly, or by a number of pseudo-random functionalities proposed previously in the literature [187], [240], [350]. A variety of feature/class binding methods are supported, including ID-Level encoding [186], vector permutation [233], and feature append-

[I] https://pypi.org/project/hdtorch/

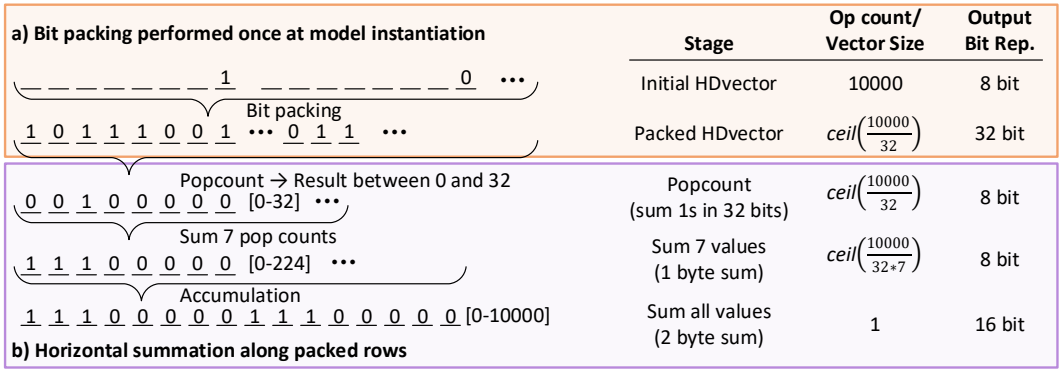

Figure 4.15: Illustration of HDTorch's bit packing and horizontal summation operations. a) Bit packing reduces hypervector memory footprint and improves bitwise operation efficiency, and b) summation enables packed hypervectors to be accumulated efficiently.

ing [305]. Pairwise similarity calculation via either hamming distance or cosine similarity [351] is supported.

HDTorch: HyperDimensional Extensions for PyTorch

While PyTorch's GPU optimization provides already excellent acceleration, its functions are not implemented with HD computing in mind and thus fail to fully optimize the HDC kernel. Namely, HD computing relies on operations between hypervectors containing 1000's of binary values. By default, PyTorch stores binary values as bytes, resulting in up to an 8x memory overhead per bit. Besides memory access, inefficiencies are also found in three key hypervector computations: namely, 1) the performance of binary `xor` operations between hypervectors during both training and inference, 2) the summation of ID-Level encoded vectors for all features, and 3) the summation along the class hypervector during hamming distance calculation. Encoding is usually the main bottleneck of HDC applications, taking up to 70% of training time [186], and is highly dependent on these inefficient operations. With this in mind, HDTorch provides four new functions with backing CUDA C++ code to address Python's shortcomings in regards to HD computing. These extensions are only applicable to binary hypervectors; bipolar hypervectors may still benefit from HDTorch's non-specific accelerations, as described later in Section 4.4.4.

Bit (Un)Packing to Improve Memory and Bitwise Operation Efficiency

The first optimization HDTorch supports is the packing of a hypervector's bits into byte blocks. This reduces hypervector memory footprint by 8x and enables bitwise operations, specifically the bitwise `xor` necessary for ID-Value encoding and Hamming distance calculation. Individual bits are packed into 32-bit integers, enabling HDTorch to take advantage of CUDA's bit counting intrinsics as described below. A complementary unpacking operation is also supported to return packed hypervectors to their original state if necessary. Both operations

are backed by CUDA code for GPU acceleration.

Horizontal/Vertical Summation for Highly SIMD Bitwise Operations

While PyTorch natively supports bitwise operations on packed bits, summation operations that can take advantage of the packed and binary nature of hypervectors are absent. Therefore, operations for performing horizontal and vertical summations are introduced to enable fast accumulations of binary vectors.

In the case of horizontal summation, CUDA's `popcount` intrinsic is utilized to count the number of bits set to 1 in a 32-bit integer. As it is known beforehand that only values of 0 or 1 are being accumulated, 8-bit summations can be used during accumulation, with an intermediate accumulation of the 8-bit accumulators into the final output summation vector every seven additions to avoid overflow.

Figure 4.15 illustrates the bit-packing and horizontal summation HDTorch operations. Vertical summation is accomplished by transposing the input bit-array before performing horizontal summation on the intermediate array. The input array is tiled into subblocks of 128x128 bits, with each subarray assigned to a CUDA warp, which is in charge of transposing the tile. Once all warps have completed their transpositions, the tiles are transposed as they are written back to the main memory.

4.4.3 Experimental Setup

To evaluate HDTorch's utility in terms of exploring the HDC design space, we perform a wide range of evaluations in terms of HDC training and inference strategies, datasets, and hyperparameter variations.

Datasets

We draw results from experiments performed on five datasets covering a range of sizes, complexities, and use cases. Four datasets are standard HDC benchmarks, while the 5th is a large, highly unbalanced medical dataset, providing a more demanding scenario than the first four datasets.

Reference HD Computing Benchmarks

To compare HDTorch performance with HDC implementations available in the literature, we use four benchmark datasets from the online-available UCI repository [356]: 1) ISOLET is an audio dataset containing spoken letters of the English alphabet, 2) MNIST is an image dataset consisting of written digits, 3) UCIHAR is a dataset for classifying human activity from smartphone inertial sensors and, 4) PAMAP is a physical activity dataset containing both inertial sensors and a heart rate monitor. We chose these datasets as they are utilized in previous works demonstrating HD computing on GPUs [186], [192], with a wide range of

Table 4.2: HDTorch benchmark Datasets (FC: feature count, CC: class count, DS: dataset size in number of samples)

Dataset	Description	FC	CC	DS [10^3]
PAMAP	Activity recognition (IMU + HR)	31	5	~80
UCIHAR	Activity recognition (Smartphone)	561	12	~10
ISOLET	Voice recognition	617	26	~8
MNIST	Handwritten digit recognition	784	10	~70
CHB-MIT	Epilepsy detection	342	2	~7056

Feature Counts (FC), Class Counts (CC) and Dataset Sizes (DS), as listed in Table 4.2.

Epilepsy Benchmark Use-Case

Beyond demonstrating HDTorch's utility on standard HD benchmarks, we wish to demonstrate its ability to enable analysis on large, computationally challenging datasets typical of real-world scenarios such as the ones for continuous monitoring of biomedical data. These datasets contain hundreds of hours of data and are usually highly imbalanced. Thus, we test the ability of HDTorch to explore HD algorithms on the CHB-MIT epilepsy dataset as well.

In previous chapters 46 features were used for classification, including mean-amplitude, entropy features and relative frequency domain features. Based on a literature review by [140] discussing the importance of various features, frequency features, as well as line-length were reportedly the most useful. Thus, here, similarly as in chapter 3.4 we keep 19 features from each of the 18 channels, calculating them on 4-second windows with a moving step of 0.5 seconds.

We organize the dataset in two manners before analysis: 1) Subsets of data for each patient that contain all seizure data and 10x more randomly selected non-seizure data (*Fact10*), and 2) the entirety of the dataset divided into approximately one-hour-long segments (*1HSeg*). Previous chapters have demonstrated that utilizing balanced seizure-to-non-seizure ratios in dataset sub-selections can lead to highly overestimated performance. Thus, in this work, we perform the first (to the best of our knowledge) assessment of HDC performance on the entire database, comparing it to the *Fact10* subset. Evaluation is performed in a time-series split cross-validation (TSCV) approach [301], where only previously acquired data can be used for training, as opposed to, for example, typical leave-one-out cross-validation. More specifically, we perform a cross-validation for each hour-long segment, with segment n as the test set and previous segments 0 to $n-1$ as the training set. This approach also reduces the runtime by approximately half, as it trains on less data in all cross-validations but the last one.

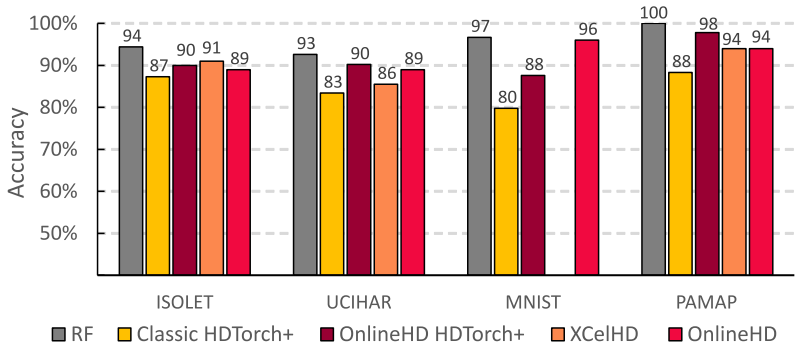

Figure 4.16: Performances of different HDC implementations, with Random Forest (RF) for reference, on 4 benchmark datasets.

Evaluation Metrics

To confirm framework correctness, we evaluate performance in terms of accuracy, memory consumption, and training and inference speedup on the four datasets described above. We perform these analyses while varying model hyperparameters, specifically, the hypervector dimension D and the online training batch size, or the frequency with which class vectors are updated as a function of the arrival of new training data.

Concerning the epilepsy dataset use-case, we analyze acceleration due to the utilization of HDTorch for encoding, training, and inference for classical and 'OnlineHD' strategies. With this realized speedup, we are able to evaluate performance of classical and 'OnlineHD' on the entirety of the CHB-MIT database. Thus, we compare episode detection accuracy for the two previously described dataset selections, *Fact10* and *1HSeg*. Performance is evaluated by concatenating predictions of all cross-validations. We perform moving average smoothing of the predicted labels with a window size of 5s, measure sensitivity (TPR), precision (PPV) and F1 score for both selections, and discuss the differences in performance.

Benchmarking Environment

Benchmarking and analysis are performed on a server system equipped with a 2-socket, 40-core Intel Xeon Gold 6242R processor capable of frequencies up to 4.1GHz and an NVIDIA Tesla V100 GPU. The software environment consists of Python v3.9.10, PyTorch v1.10.2, and the CUDA driver v11.6. Profiling is accomplished via PyTorch's native profiler, capable of profiling CPU and GPU runtime/memory consumption by function.

4.4.4 Results

The following sections detail the results of our experiments. For clarity, results utilizing hypervector CUDA extensions are marked as HDTorch+ in the following figures.

Model Accuracy Analysis

Figure 4.16 shows the accuracy of online and classical HD model implementations with respect

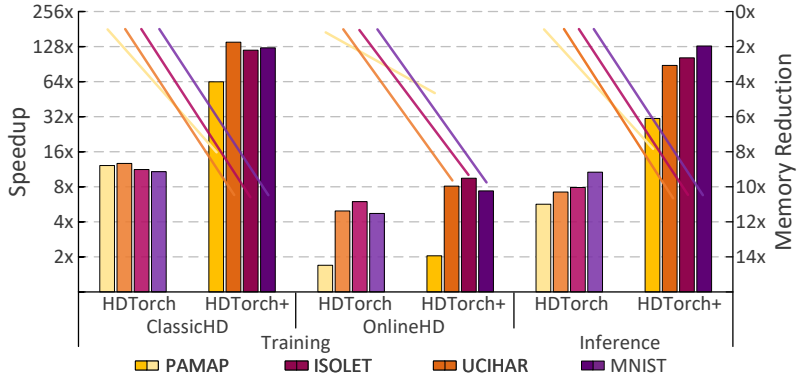

Figure 4.17: Speedup comparison between different HD learning implementations. 4 different benchmark datasets are used.

to random forest performance. We also compare the 'OnlineHD' approach with two implementations found in the literature. *XCelHD* [186] uses a modified TensorFlow implementation of the online HD workflow described in Section 1.3.2, and *'OnlineHD'* [192] uses an original HD floating point model with a non-standard encoding approach implemented in PyTorch.

Online training improves performance in comparison to classical HD for all datasets. Furthermore, our online HD implementation using HDTorch is similar in accuracy to the implementations found the literature. It should be noted that random forest outperforms all HD computing implementations, indicating the necessity for further optimization of HD computing to reach state-of-the-art accuracy results. This situation further motivates the need for efficient design space exploration of HDC algorithms.

Time Performance Analysis

Figure 4.17 illustrates runtime accelerations achieved by HDTorch/HDTorch+ in comparison to HDTorch run on the CPU. Acceleration for training on both classical and online HD, and acceleration for inference (equivalent for both approaches) is illustrated. Comparison with XCelHD [186] speedup was not possible as the code is not publicly available. As can be seen, both HDTorch and HDTorch+ greatly reduce benchmark runtime: HDTorch provides up to (12.7x/6x)/10.7x (training)/inference speedup for classical/online runtime, respectively, while HDTorch+ improves these gains to (139x/9.5x)/130x. Note that 'OnlineHD' speedup is significantly lower as a result of the necessity to re-calculate the class vectors after every datapoint, a challenge related to batching of the data. Finally, the observed differences in speedup for different datasets is due to the different dataset feature counts; for example, PAMAP has the smallest number of features and thus the smallest possible speedup due to more limited parallelization opportunities.

Memory Consumption Analysis

Figure 4.17 also illustrates trends in memory consumption across the datasets. Memory values are normalized to the memory usage of running training and inference on the CPU. It

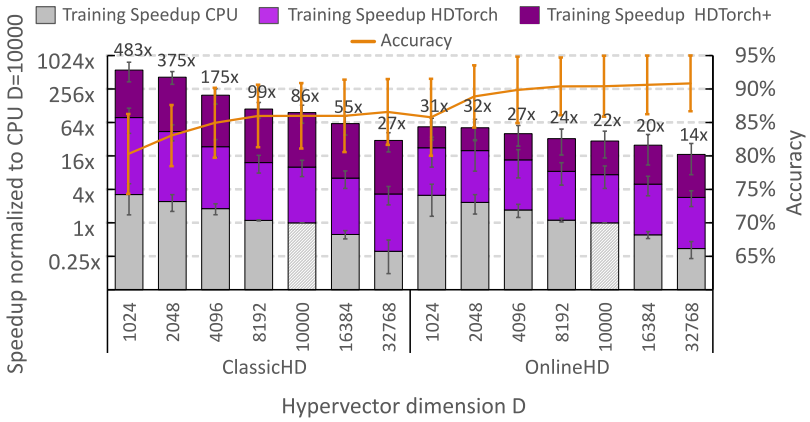

Figure 4.18: Average speedup for HDTorch-CPU, HDTorch and HDTorch+ for different dimensions of HD vectors on 4 benchmark datasets. Values are normalized to the runtime of the HDTorch model running on CPU with the typical D=10000.

can be seen that utilizing HDTorch without HDTorch+ extensions does not reduce memory usage, as the tensors used to store hypervectors are identically sized on either the CPU or GPU. Introducing HDTorch+ extensions, on the other hand, reduces memory consumption by approximately 10x for both training and inference in the aforementioned benchmark datasets. This is due to reducing the memory footprint for hypervectors by 8x by bit packing, and utilizing the more memory efficient HDTorch+ bit-array summation functions, which instantiate fewer intermediate tensors during computation. Once again, PAMAP is an outlier on memory consumption reduction as it has an order of magnitude fewer samples compared to the other datasets.

Influence of the parameters

We also evaluate the impact of hyperparameter variance on accuracy and runtime for the four benchmarks. We find a trade-off between model complexity/size, accuracy, and runtime. Measured runtime and accuracy values are averaged across the four datasets, and error bars indicate the standard deviation of the averaged values.

Figure 4.18 illustrates the impact on model training time when varying the width D of the class and feature hypervectors between 1024 to 32768 bits. The runtime values are normalized to the runtime of the HDTorch model running on CPU with the typical D=10000. As can be seen, reducing D drastically reduces runtime, up to 483x/31x for classical/online learning with HDTorch+, at the cost of an average 6% decrease in accuracy. On the other hand, increasing D past 10000 provides little improvement to accuracy. However, our results show that even a model with a D=32768 runs 27x/14x faster over a CPU model of D=10000 for classical/online HD, indicating that if a user would like to test a higher dimensional model on their dataset, HDTorch makes this feasible in a reasonable amount of time.

'OnlineHD' benefits less from HDTorch acceleration due to the fact that 'OnlineHD' is trained sequentially, with one data sample accumulated into the class vector at a time, and thus cannot be highly parallelized. There is, however, the possibility to improve runtime by only updating class vectors after a batch size of n new datapoints are received. Figure 4.19 illustrates

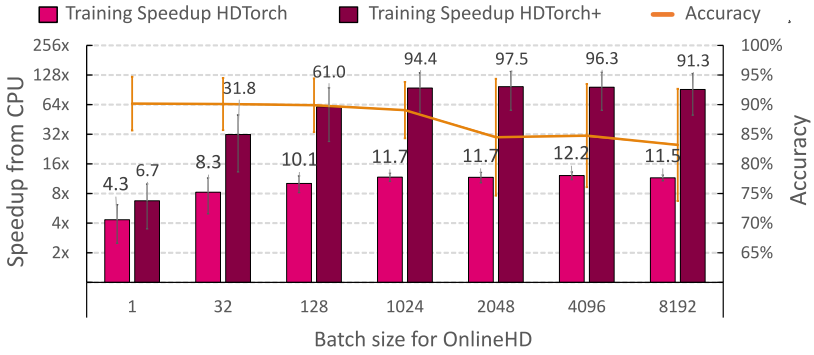

Figure 4.19: Acceleration with respect to HDTorch-CPU, compared for HDTorch and HDTorch+ implementations. Comparison is given for training using different batch sizes.

batching effects on model runtime and accuracy. It can be seen that batching data significantly decreases training time, especially for HDTorch+, with an average of 68x performance gain achieved across all batch sizes and datasets. This comes at a cost of an average accuracy drop of 7% at a batch size of 8192. This drop will vary according to dataset complexity; hence, batch size should be tuned to the particular use case of the model.

Epilepsy Detection Use Case

Finally, we test classical HD and 'OnlineHD' implementations on a real-life dataset for epilepsy detection. Figure 4.20 illustrates HDTorch and HDTorch+ speedup with respect to CPU-HDTorch for classical/online HD. Accelerations are shown for the encoding, training, and inference stages. For classical HD, HDTorch achieves a 7x speedup for all stages, with HDTorch+ providing an additional 10x gain for each stage, for total speedups of 79x, 78x, and 70x for encoding, training, and inference, respectively. For online HD, performance gains are similar for encoding and inference, whereas, for training, gains are significantly reduced, to 2.1x and 2.7x for HDTorch and HDTorch+. This is due to the parallelization constraints described in Section **??**.

The speedup achieved by using the HDTorch library enables us to evaluate HDC performance on the entirety of the CHB-MIT database. Evaluation of HDC performance on such a big database enables better insights into the performance of HDC algorithms for real-life applications on wearable devices. Figure 4.21 illustrates the detection of epileptic episodes for the two dataset organizations, *Fact10* and *1HSeg*. For *Fact10*, there is no significant difference between performance of random forest, classical HD and 'OnlineHD', with all of them detecting almost all seizures with some amount of false positives. On the other hand, training and predicting on the *1HSeg* dataset organization shows a significant decrease in performance for classical HD. While it still detects almost all seizures, it also has significantly more false positives, thus reducing the F1 score. While random forest and online HD miss some seizures, they also avoid such a significant drop in precision and F1 score. Even so, random forest slightly outperforms 'OnlineHD' performance.

These results indicate two key takeaways. First, analyzing a subset of data may not generalize

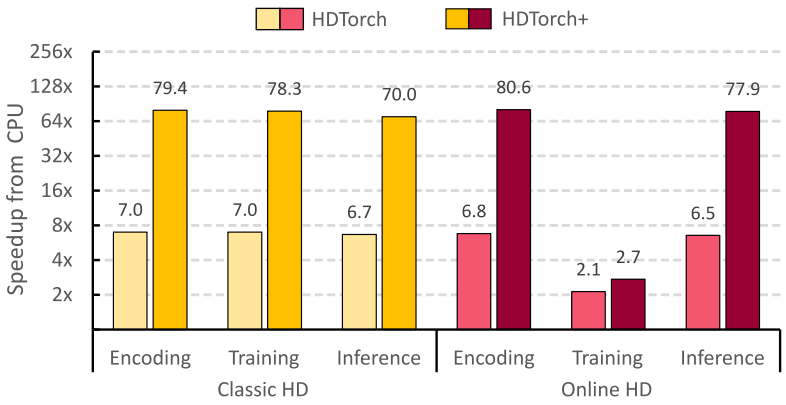

Figure 4.20: Acceleration in respect to HDTorch-CPU, compared for HDTorch and HDTorch+ implementations, for encoding stage, training and interference on CHB-MIT dataset.

to performance on the entirety of a dataset, especially when the dataset is large and highly unbalanced, as is the case of the CHB-MIT database. Second, new training strategies are constantly improving HD computing's accuracy, bringing it closer to that of other standard ML models, such as random forest. Both takeaways motivate the necessity for HDTorch, which enables new HDC strategies to be explored on realistic datasets, paving the way for future breakthroughs in the field.

4.4.5 Conclusion

In order to bring HDC performance in line with state-of-the-art ML algorithms and position it as an algorithm for continuous online monitoring of wearables in healthcare, algorithm optimization, and design space exploration are necessary. Thus, in this work we have presented HDTorch, the first open-source, PyTorch-based library for HD computing with CUDA extensions for hypervector operations.

On four HDC benchmark datasets, we demonstrated an average 111x/68x training speedup for classical/online HD, respectively, and an average 87x speedup for inference. In addition, we have shown that HDTorch's CUDA extensions for HDC operations reduce training/inference memory consumption by up to 10x. HDTorch's utility and flexibility have been demonstrated by analyzing the effects of hypervector dimension and batch size on model accuracy and runtime.

Finally, the speedup achieved by using the HDTorch library enables us to evaluate HDC performance on the large, highly unbalanced CHB-MIT epilepsy dataset, where we demonstrate that the performance resulting from typical approach of training on a subset of the data does not necessarily generalize to training on the entire dataset. This important observation must be carefully considered when developing future HD models for medical wearable devices.

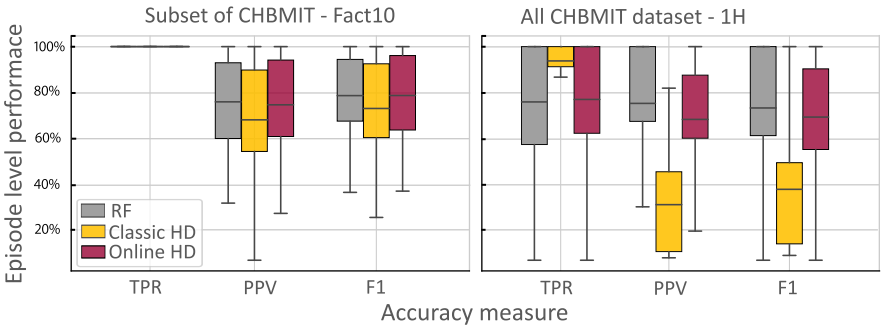

Figure 4.21: Epilepsy detection performance comparing training on a subset of data (typical in previous literature) against using the whole CHB-MIT dataset. Boxplots represent performance distribution for all 24 subjects, with mean performance marked as a horizontal line.

4.5 Conclusion

This chapter discusses the ability of HD computing to deal with large and also highly unbalanced datasets. In the introduction, we discuss methodological choices that need to be considered when designing the experiment. Then in section 4.2, we propose a new learning approach called 'multi-centroid' learning, while in section 4.3, we compare it with other learning approaches from the literature. In the end, in section 4.4, we propose an open-source library to train HD models more efficiently.

More specifically, in section 4.2, we demonstrated that the proposed multi-centroid approach has significantly improved performance when compared to a simple single-centroid HD model. More importantly, the improvement was bigger for more challenging, i.e., more unbalanced datasets. With the multi-centroid approach, when increasing the portion of non-seizure data in the dataset, performance did not change significantly, demonstrating the resistance of this approach to the imbalance in data. Similarly, in the first step of the proposed algorithm, when new subclasses are created, a highly variable number of seizure and non-seizure subclasses for each subject, reflected the complexity of the data and the classification challenge itself. However, after the step of reducing the number of sub-classes, on average, one to three centroids per class were needed. Interestingly this number did not change much with increasing unbalance of the dataset. Thus, it seems that the multi-centroid approach can be an important step forward to achieving high performance in epilepsy detection with real-life data distributions, where seizures are infrequent, especially during online learning.

In section 4.3, first, we have shown a significantly lower performance of the standard HD approach compared to the random forest one, on unbalanced data subset, which is mainly due to many false-positive seizure detections. Then we demonstrated that enhancing this standard HD approach by using multi-pass, multi-centroid learning and their combination significantly improves the performance. However, only their combination reached a performance level comparable to the random forest. 'OnlineHD' also significantly improved the performance but didn't reach statistically no difference from RF performance. However, when analysing

memory and computational complexity of all approaches, 'OnlineHD' approach is much more memory and computationally friendly when compared to a multi-pass approach, which takes more time to train, or when compared to a multi-centroid approach, which requires more memory. Nonetheless, this analyses and results have proven the potential HD computing has for real-life epileptic seizure detection.

Finally, in order to bring HDC performance in line with state-of-the-art ML algorithms and position it as an algorithm for continuous online monitoring for wearables in healthcare, algorithm optimization and design space exploration are necessary. Thus, in section 4.4 we have presented HDTorch, the first open-source, PyTorch-based library for HD computing with CUDA extensions for hypervector operations, and have shown significant speedup both for training and inference of classical and 'OnlineHD' computing.

Limitations

Similarly, as in previous chapters, due to the slow training with a large dataset, still the majority of the analysis in this chapter was also done using a subset of the CHB-MIT dataset, namely a subset with only 10x more non-seizure. It would be interesting to test multi-centroid approach as well as compare all approaches using the full CHB-MIT dataset. Moreover, other datasets could be used too.

We believe that all proposed improvements for HD learning, namely, *multi-centroid*, *iterative* and *weighted ('OnlineHD')* learning, tackle the problem in different ways and that finding ways to combine two of them or even all three approaches could lead to even better learning capabilities. Thus, we think this is one of the future venues for the research.

5 Personalized nature

Seizures show highly personalized patterns, leading to personalized models outperforming general models. However, the need to collect enough data to be able to train personalized models significantly limits potential wearable outpatient applications. An ideal scenario with one general model for all individuals that is preloaded on a device and performs well is not realistic. New methods of personalizing general models using the characteristics of individual patients [325] are needed.

As mentioned previously, other aspects that need to be considered when designing algorithms for future lightweight wearable devices with long battery life are computational complexity and memory consumption. Many state-of-the-art ML algorithms for epilepsy detection are, for this reason, not ideal for wearable devices. Thus, hyperdimensional computing is posed as a potential alternative.

However, most of the HDC models developed by researchers are either personalized (only using patient-specific data) [206], [240], [269] or generalized (using all patients) [204] and are rarely discussed or compared. To clarify the terminology defined in the thesis, under 'generalized', we do not talk about the detection or prediction of generalized seizures but the ML approach, where models are not personal but instead trained on data from multiple subjects. There is no research on the topic of interplay between generalized and personalized models. However, the ability to compare them could enable interesting insights about individual subjects but also the general epilepsy population or dataset itself. Thus, in this chapter, we want to demonstrate several interesting aspects of HD computing models, such as the ability to create generalized models from personalized ones, compare them, and also to build hybrid models from personalized and generalized ones at the same time.

This chapter contributes to the state-of-the-art in the following ways:

- We show how HD computing models represented as hypervectors can be used to compare inter-personal seizure and non-seizure models.
- We present several approaches to create generalized models from personalized models

of individual subjects. Then, we study how many subjects are needed to achieve stable generalized models.
- We compare the performance of epilepsy detection between personalized and generalized models, aggregated for all subjects, and also for individual subjects. Furthermore, we design hybrid models that rely both on personalized and generalized models and show that they can achieve better detection accuracy than just generalized or personalized models.
- Finally, we test knowledge transfer between models of two EEG epilepsy datasets, one big but short-term database (Repomse [272]), and one long-term but with a smaller number of subjects (CHB-MIT [242]).

5.1 Personalized nature of epileptic seizures

Epileptic seizures are a highly personalized and variable phenomenon, with different individuals experiencing seizures with different frequencies, duration, intensities, and triggers. The authors of [21], discussed the complex interplay between genetic, environmental, and behavioral factors that contribute to the personalized nature of epilepsy and finally highlighted the importance of personalized treatment approaches that take into account individual differences in seizure characteristics and underlying etiology.

In [357], more than 1 million seizures were logged from more than 10000 subjects (56.7% children) between 2007 and 2016. The study found a range of interesting seizure characteristics among different etiologies, temporal patterns in seizure fluctuations, and specific triggers. Children had more frequent seizures than adults did, with a median monthly seizure frequency of 3.5 vs. 2.7. Further, seizures showed to be correlated with circadian patterns: there was a higher frequency of seizure occurrence between 7 am and 10 am, as well as during the weekdays when compared to the weekend. Moreover, longer seizures (>5 or >30 min) were more often caused by triggers related to irregular sleep and tiredness, emotional stress, or bright and flashing lights.

This variability and complexity of seizures is directly visible in EEG patterns. Thus, in Fig. 5.1, we show different seizure segments from the same person (subjects 2 and 4) from the CHB-MIT database. It is clearly visible how seizure signals can show a very different morphology between different channels of the same seizure, as well as between different seizure instances of the same person. For example, some patients show multiple seizure onset sites that each produce their own characteristic seizure dynamics [358]. Thus, even within the same patient, ictal onset patterns [359], the extent of seizure spread [360] or seizure recruitment patterns [361] can also differ.

Although several studies [362]–[364] have developed methods to quantitatively compare seizures intra- or inter- patient, the current gold standard for seizure comparison remains visual inspection of ictal EEG by trained clinicians. Visual inspection is extremely time-consuming and subjective and can miss important features, including functional network

interactions, that are difficult to detect visually [365]. Thus in the future, more advanced algorithms for quantifying seizure similarities as well as seizure origins are needed.

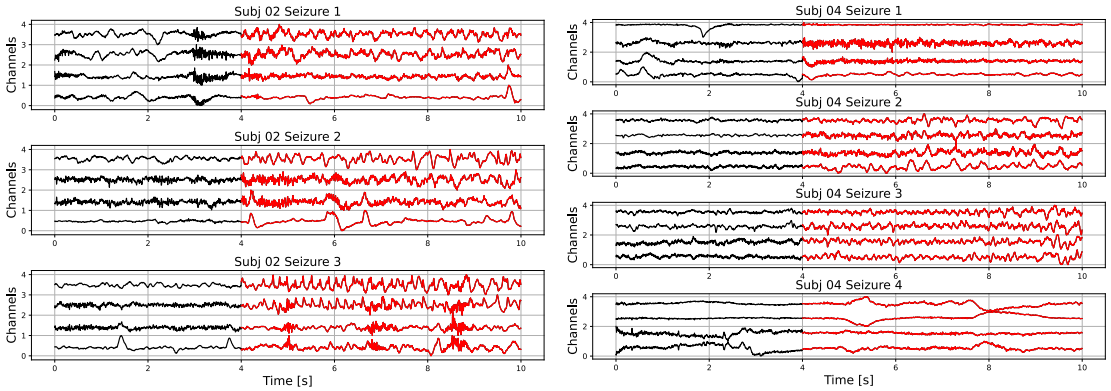

Figure 5.1: Raw signal showing several seizures from subjects 2 and 4 of the CHB-MIT database. Only the first four channels are shown.)

In a recent paper [325], authors visually identified patients' specific EEG signatures per patient based on the intra-individual stereotypy of ictal pattern. The method showed to be able to dramatically minimize the false detection of ictal EEG. As surprisingly little is known about whether and how seizures vary in the same patient, in another paper [365], authors compared within-patient seizure network evolution using intracranial EEG recordings of over 500 seizures from 31 patients with focal epilepsy. They found that seizures with similar patterns tended to occur closer together in time, pointing towards the fact that various modulatory processes, operating at different timescales, shape within-patient seizure evolution. These resulting variable seizure patterns also require tailored treatment approaches.

Indeed, most of the machine learning approaches for epilepsy detection in the literature use personalized models [73], [206], [240], [269], [350] rather than one generalized model for all subjects. Even if general models would be more appropriate in practice, their performance is still too low for their application.

Individual differences are not only visible in different patterns of epileptic seizures but also in response to the therapies [366]. In the past three decades, more than 20 new anti-seizure medications have been developed [366], but still, one-third of patients experience recurrent seizures. Thus, the precision medicine approach for the treatment is strongly advocated.

5.2 HD computing for personalized and generalized epilepsy models

Due to mentioned practicability of generalized models, but also the strong personal character of epilepsy, here we investigate and propose how HD computing can be used to study the interplay between generalized and personalized models. In the end, we also propose how these models can be combined to form hybrid models.

5.2.1 Comparing models

HD computing training results in models which are in the form of long hypervectors. For example, for epilepsy, typically, we have one model (vector) for ictal (seizure, S) and one model for inter-ictal (non-seizure, NS) data. These two vectors can be easily compared using the cosine or Hamming distance, giving us the value of separability/similarity of the two classes. If we train only using data from individual seizures, comparing resulting vectors can quantify the similarity between sets of seizures (intra-subject similarity). This can further help to quantify how repeatable the seizure patterns of specific patients are.

Similarly, performing personalized training results in pairs of seizure (ictal) and non-seizure (interictal) model vectors for every subject. Then, an interesting next step is to study the difference between models of different subjects. This comparison leads to a characterization of the similarity of seizure patterns between different patients (ictal-ictal or S to S), the similarity of non-seizure data between patients (interictal-interictal or NS to NS), as well as inter-subject ictal-interictal (S to NS) similarity. These values can help us to understand how patients could take advantage of generalized models from other patients. Potentially, it can even help to group patients depending on the similarity between their seizure or non-seizure models.

5.2.2 Creating generalized models

Focusing on generalized models, they can be created by training on the whole data containing all subjects at once, but they can also be created from already trained personalized models of individual subjects. This approach is very interesting for distributed learning where, for example, each subject's model is created on a different wearable device and without sharing sensitive clinical data. Generalized models can be combined on a central server where all individual patient models are sent. It is very important to highlight that these personalized models are privacy-preserving, as it is impossible to decode back individual data after it was encoded and accumulated to hyperdimensional models.

We test and compare different approaches to build general models from personalized ones: 1) *Avrg*, 2) *WSub* and 3) *WAdd&Sub*. The simplest *Avrg* approach is based on the average of all personalized model vectors (and normalize to have binary model vectors again), as defined in equation 5.3. $VecP$ stands for personalized vector of individual subject (whether it is seizure or non-seizure model vector), and $NumS$ stands for number of subjects that are available in a dataset. Another approach to creating generalized models is similar to *'OnlineHD'* training, where before adding a new subject, its model vectors are compared to the current generalized

models and added proportionally to the novelty they would bring. More specifically, the personal vector of the correct class is added, but the opposite class is also subtracted and multiplied with the weight. This is defined in equation 5.4. We call this approach *Wsub* as only the opposite class is multiplied with the weight. The third tested approach is called *Wadd&sub* as defined by equation 5.5. Here the correct class is also multiplied with the corresponding weight before being added to the generalized model vector. In the end, we also test if there is any benefit of performing weighted adding in an iterative manner. In all cases, the weight for adding the correct class is defined as

$$w_{corr} = \alpha(1 - hammDist_{corr}) \tag{5.1}$$

so that the more similar (larger Hamming distance) personalized and generalized vectors are, the smaller the weight is with which it is multiplied, as it doesn't bring a lot of new information. On the other hand, weight for the opposite/wrong class is defined as

$$w_{wrong} = \alpha * hammDist_{wrong} \tag{5.2}$$

so that if the opposite class personal vector is very similar to the current generalized vector (Hamming distance is high), weight is also high, reducing that vector significantly. In both equations for weight, there are additional factors α_{corr} and α_{wrong} that define the importance of adding the correct class or subtracting the wrong class. In this case, we use value 1 for both of them, but they could be further optimized in the future, depending on the specific use case.

$$VecG = \frac{\sum_{p}^{NumS} VecP_{corr}}{NS} \tag{5.3}$$

$$VecG = \frac{\sum_{p}^{NumS} VecP_{corr} - w_{wrong} VecP_{wrong}}{\sum_{p}^{NS} 1 - w_{wrong}} \tag{5.4}$$

$$VecG = \frac{\sum_{p}^{NumS} w_{corr} VecP_{corr} - w_{wrong} VecP_{wrong}}{\sum_{p}^{NS} w_{corr} - w_{wrong}} \tag{5.5}$$

Further, we can also track the evolution of generalized models as more subjects are added. For example, tracking average similarity with personalized models as we add more personalized models enables us to determine how many subjects are needed to reach stable generalized models that do not change significantly when adding more subjects.

5.2.3 Hybrid models

Using generalized models would be most convenient from a practical perspective. Models could be trained once and then used on all new subjects. This is particularly interesting for wearable outpatient applications. But as epilepsy exhibits highly personalized patterns, generalized models usually do not perform well enough. Also, real-life recordings are highly unbalanced, which poses another challenge for good performance, as shown in [196].

Having both personalized and generalized models, we can compare the detection performance on individual subjects. We can further divide patients into groups for which personalized perform better, or for which generalized models are better. Here we investigate the hybrid approach of using generalized models by default and replacing them with personalized models if the generalized performs worse than a certain threshold. We study the percentage of subjects that would need to use personalized models depending on the chosen performance threshold of generalized and how the overall performance over the whole population changes when including more subjects using generalized model.

Finally, HDC models allow us to have even more flexibility, for example, by combining both personalized and generalized models at the same time, rather than choosing only a personalized or a generalized model. More specifically, since seizure and non-seizure models are just simple hypervectors, we test the performance when only the seizure model is kept personalized, while the non-seizure model vector is kept generalized (*NSgen-Spers*), or the other way around (*NSpers-Sgen*). The *NSgen-Spers* approach is interesting as it reuses non-seizure models from other subjects where only seizure data from individual subjects are needed to create their personal seizure models (*NSgen-Spers*), potentially saving a lot of time for training personalized models. On the other hand, the *NSpers-Sgen* model is very interesting since we only require non-seizure data of each subject to have a personalized model, which is much easier to gather than seizure data. This would mean that the new subject would only need to record some amount of non-seizure data rather than recording for days or weeks to gather sufficient amount of seizure data. In this work, we compare the performance of both of these approaches with personalized and generalized model performances.

5.2.4 Models transfer between databases

In the literature, knowledge transfer between models of different datasets is rarely tested. In the recent paper [150], deep learning approach called M2D2 ('Maximum-Mean-Discrepancy-Decoder') was used for automatic temporal localization and labeling of seizures in long EEG recordings of the Epilepsiae [26] and CHB-MIT databases. Their problem formulation is slightly different than ours, detecting windows of a certain size with a high probability that they contain seizures, intending to simplify and speed up the labeling process of a new database. But overall, this is one of the few works that test models trained on a different dataset.

For hyperdimensional computing, based on our knowledge, there is no work that performed generalized training so far or that further combined models from several datasets. Here, HD

computing enables us not only to use generalized models from different datasets but also to create hybrid models where only one part of the model is from a different dataset. For example, we will test when only the seizure (or non-seizure) model is from a different dataset. This enables us to better understand the knowledge transfer between models of two datasets, in both directions.

5.3 Experimental setup

Datasets

The main analyses were performed using the Repomse database [272] curated by the Lausanne University Hospital. All the details about dataset preparation are explained in Sec. 2.1.2. After rejecting specific subjects due to various reasons (see Fig. 2.2), we kept 286 subjects, which is still a much larger number than we could have get from any other publicly available dataset, and then it was ideal for studying generalized models. Original recordings contain a single seizure with a maximal length of 1 min, with usually 3 minutes of interictal recording before the seizure onset. Similarly, as in [280] we exclude 1 min of pre-ictal, and utilize 1 min of inter-ictal data. In this way, every file contains a balanced amount of seizure and non-seizure data.

Finally, to test knowledge and models transfer between databases, we also used the public CHB-MIT database [242]. We kept same 18 channels as in Repomse database. From the raw database, similarly as in [99] we prepared a dataset that contains ten times more non-seizure data than seizure data to be closer to a more real-life data balance. The main reason to avoid a balanced scenario, which is common in many works in the literature, is that it can lead to a highly overestimated performance, not achievable during continuous monitoring with a wearable device [196]. Non-seizure segments were chosen randomly from available non-seizure data, but excluding data 1 min before and 15 min after a seizure, and arranged around seizures. Thus data structure was similar to Repomse, with one-seizure-per-file but with a bigger amount (10x more) of non-seizure data.

Features used

We use the mean amplitude and both relative and absolute power spectral density in the five common brain wave frequency bands: delta: [0.5-4] Hz, theta: [4-8] Hz, alpha: [8-12] Hz, beta: [12-30] Hz, gamma: [30-45] Hz, and low-frequency components: [0-0.5] Hz and [0.1-0.5] Hz. We also included the line length feature [96] and the 'approximate-zero-crossing' features (AZC), as proposed in [302]. The AZC features are based on simplifying the EEG signals by applying a polygonal approximation to mimic how our brain selects prominent patterns among noisy data. Then, a simple zero-crossing count is used as an estimation of the dominating frequency of the signal. We extracted six features by using six different thresholds for polygonal approximation. Thus, in total, we extract 25 features from data segmented into 4-second windows with a 0.5-second step. Before extracting the AZC features, the data is filtered with a 4th-order, zero-phase Butterworth bandpass filter between [1, 20] Hz.

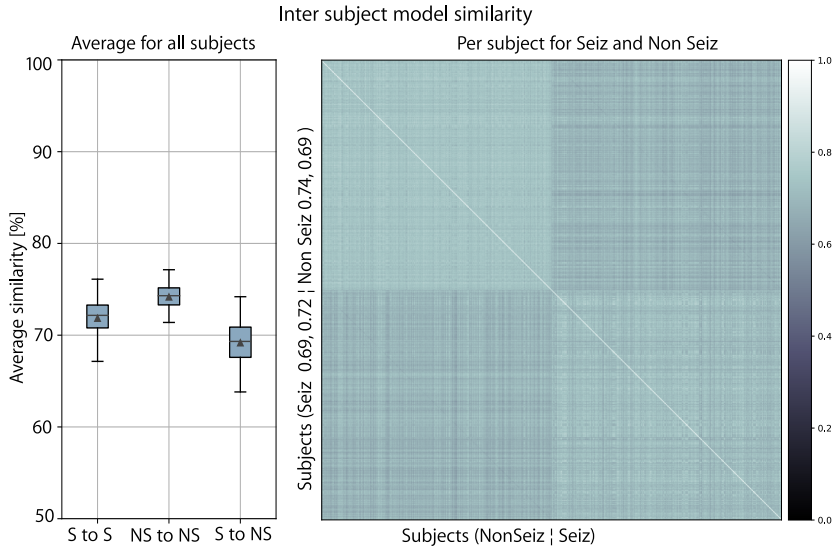

Figure 5.2: Inter-subject similarity between model vectors of individual subjects (both between ictal (Seiz, S) or inter-ictal (NonSeiz, NS) models).

Training and evaluation

For the HD computing workflow, we can choose between the standard learning or *'OnlineHD'* approach. Since in [99] authors compared standard and *'OnlineHD'* for the use case of epileptic seizure detection and showed better performance of an *'OnlineHD'*, we use it in this work as well. Due to the subject-specific nature of epileptic seizures, and one-seizure-per-file division, training and evaluation of personalized models were done using leave-one-seizure-out cross-validation. In the end, all test predictions are appended into the original time order, and performance is calculated. This is a training method that is more appropriate for long-term datasets, without excluding any data and thus leading to more representative performance for real-life applications [367]. For generalized models, on the other hand, and for both databases, the leave-one-subject-out cross-validation approach was used. To report the overall performance, we calculate the average over all subjects, both for personalized and generalized approaches.

The performance of the classifier is evaluated with respect to episode detection and duration-based detection, measuring sensitivity, predictivity, and F1 score. The performance at the episode level groups the signal into blocks of seizure and non-seizure, as detailed in [368], while the performance at the duration level considers the correct prediction of each sample. In epilepsy detection, raw label predictions often lead to unrealistic seizure dynamics (e.g., seizures lasting only a few seconds or seizures that are only a few seconds apart). Thus, label post-processing is an integral part of the pipeline. We use Bayesian post-processing that calculates cumulative probability of classes and thresholds them as described in [280]. Threshold value of 1.5 and window length of 5s were used.

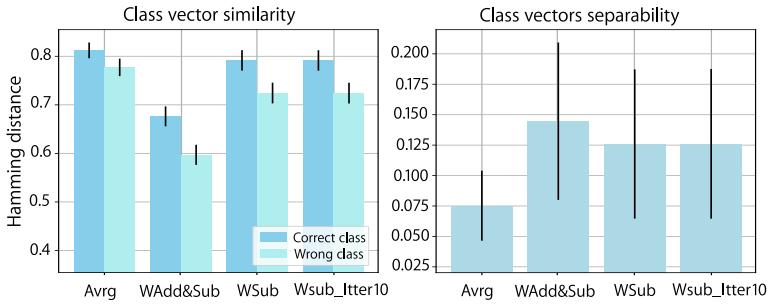

Figure 5.3: Comparing different approaches to creating generalized models from personalized ones. Similarity between seizure and non-seizure model vectors as well as overall class separability is measured.

5.4 Results

First, we show how HD computing can compare seizure and non-seizure models of individual subjects. Then we show results related to the creation of generalized models, followed by a comparison of the performance of personalized and generalized models. Finally, we show the possibilities of hybrid models and models trained on different datasets.

5.4.1 Inter-subject similarity

Fig. 5.2 shows a comparison between individual seizure and non-seizure models of 286 subjects. Non-seizure to non-seizure (NS to NS) models are the most similar, followed by seizure to seizure (S to S). Finally, seizure-to-non-seizure (S to NS) models are the least similar, which is expected. There is a significant difference between *S to S* and *NS to NS*, and also there is a significant difference between *S to NS* similarities and *S to S* as well as *NS to NS*. Wilcoxon paired tests were performed to compare distributions, with all $p-$values being lower than 10^{-15}. A similar comparison can be made within the data of one single subject (*intra-subject*), comparing models of individual seizures.

5.4.2 Creating generalized models

Fig. 5.3 shows the quality of generalized models created with four different approaches. We measure the similarity between the generalized and all individual personalized models, showing the mean and standard deviation. The similarity is measured for all options: generalized seizure to personalized seizure models (*S to S*), non-seizure to non-seizure (*NS to NS*), which represent similarities between correct classes, and similarly *NS to S* and *S to NS*, representing similarities between opposite classes. From similarities between correct and opposite classes, we further calculate class vector separability as a measure of how separable the models are.

When comparing the four approaches, the simplest averaging (*Avrg*) of individual subject models results in the highest similarity between the correct classes but also a high similarity with

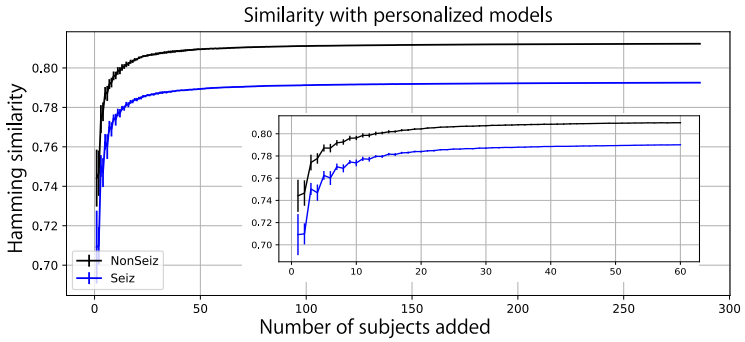

Figure 5.4: Evolution of generalized vectors as adding one by one individual subject. Average similarity with all personalized subjects is measured to characterize how stable are generalized vectors. The inner plot shows zoomed part of outer plot.

the opposite class, leading to the lowest class separability. When we add weighted subtraction of opposite class vectors in the *Wsub* approach, both similarities drop, but opposite classes become less similar, leading to better separability. Finally, if adding the correct class individual vectors is also multiplied with weight (proportional to the novelty of individual subjects vector) accompanied with weighted subtraction of opposite class (*WAdd&Sub*), similarities drop significantly but lead to the highest classes separability. Since the order in which subjects were added could have influenced the similarity of generalized models with personalized ones that were added at the beginning of the process, we also tested iterative creation, where we passed through all subjects several times. As shown in Fig. 5.3 this didn't change any of the measured values.

Next, we tried to assess how many individual subjects are needed to get stable generalized models. Thus, we measure how similar the generalized model is on average to all individual personalized ones (no matter if they were already added to the generalized or not). We repeated this 10 times, adding subjects in a different order, and we show the average similarity in Fig. 5.4. A clear trend of similarity plateauing after adding approximately 40 patients (both for seizure and non-seizure vectors) quantifies the minimum number of patients needed to create stable generalized models.

5.4.3 Personalized vs. Generalized models

First, we were interested in comparing performance between personalized and generalized models. If we look at the F1 score for episodes, personalized models were better for 95 out of 286 subjects, generalized were better for 152 subjects, and for 39 subjects, both models had the same result. In Fig. 5.5, we compared the performance of generalized and personalized models in three cases: the average of all subjects, the average of subjects where generalized models performed better (F1 score for episodes), and the average of subjects where personalized models performed better. The figure shows that for the whole group population, generalized performed, on average, slightly better than personalized models in terms of F1 score for

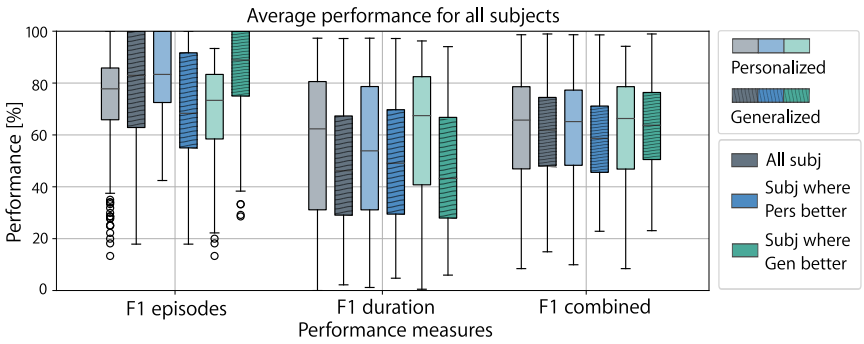

Figure 5.5: Comparing performance of personalized and generalized models: 1) for all subjects (in gray), 2) on subjects where personalized models perform better (in blue) and 3) on subjects where generalized perform better (in green). Performance without any postprocessing is shown, for F1 score on the level of episodes, duration and gmean of both (F1 combined).

episodes (F1E), but when looking at the F1 score for the duration (F1D), where we care about the classification of each sample, personalized models performed much better. When observing only subjects where personalized performed better, it can be seen that, indeed, F1 score for episodes (F1E) dropped significantly for generalized performance. Similarly, for subjects where generalized performed better, F1E was much higher for generalized than personalized models. This gap between performance of personalized models and generalized models for individual subjects opens the possibility to stratify patients into two groups.

As explained before, for real-life applications, ideally one generalized model that would perform well for all subjects would be the best solution. But as shown, generalized models do not always perform the best. Thus, we tested an approach in which we used personalized models in case the generalized model performed under a chosen threshold. We measured how overall performance for all subjects depends on the chosen threshold and the final percentage of generalized models used. In Fig. 5.6, results show that the highest overall performance (average of all subjects in the Repomse database, if looking at F1E) is achieved when 60 to 80 % of subjects used generalized models. With full horizontal lines, the hypothetical performance with an optimal selection of personalized or generalized models among subjects. This is of course not achievable in practice, because we cannot know which model performs better if both models are not built and tested. Thus, here we use an approach where by default generalized model is used, unless it performs worse then chosen threshold. More precisely, if the minimal satisfactory performance threshold of generalized models is set to 60% this would result in 80% of patients having generalized models. If we want to be more strict and choose a performance threshold of 75% we will use 60% of generalized models. On the right graph in Fig. 5.6, we show the performance after moving average post-processing, and we can see that even if for raw predictions the ideal performance (full horizontal lines) were not reached, with post-processing it is possible to reach it.

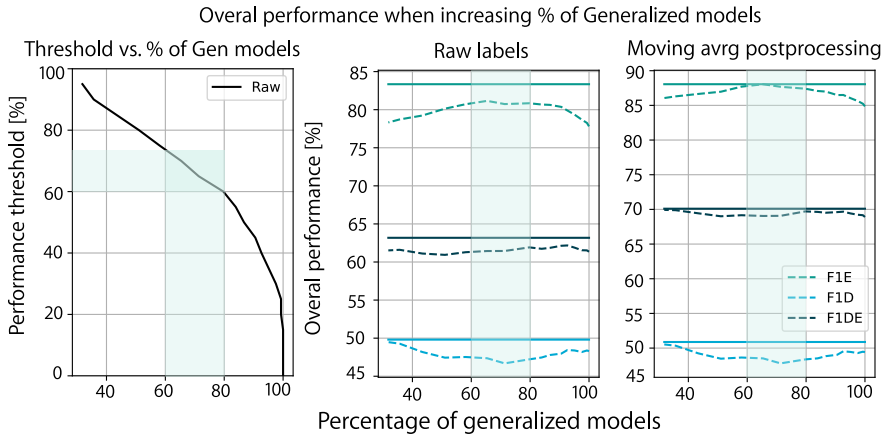

Figure 5.6: We investigate combinations of generalized and personalized models. Namely, by default for all subjects generalized models is used, but if it performs worse then certain performance threshold, personalized models is used. On the left graph percentage of subjects that would use generalized model is shown as threshold is decreased. On next two graphs, overall performance when changing percentage of generalized models is shown.

5.4.4 Hybrid models

Finally, a unique possibility of HD computing models is to actually not only choose a personalized or generalized model per subject, but the ability to combine and create hybrid models. Fig. 5.7 shows the performance of random forest, personalized, generalized and two hybrid models (*NSgen-Spers* and *NSpers-Sgen*). Sensitivity (TPR), precision (PPV), and F1 score for episode and duration based performance are shown. When looking at episode performance, HD computing models outperformed random forest with the same set of features. Further, generalized and personalized models perform similarly. This is probably due to the fact that the Repomse dataset is a balanced dataset with not so many non-seizure data making generalized classification easier on the episode level. Hybrid *NSpers-Sgen* models achieves slightly higher F1 score for episodes. This is due to higher precision (PPV), or less false positives.

If we look at performance on the level of duration, it can be noticed that generalized and *NSpers-Sgen* models perform significantly worse in terms of sensitivity (TPR), leading also to a lower final F1 score. The *NSgen-Spers* approach, on the other hand, performs as well as personalized models in terms of sensitivity, having also a slightly higher precision and finally a slightly higher final F1 score (F1D) than personalized models. This means that non-seizure data can be used from other subjects but that for good sample-by-sample classification it is important to keep seizure models personalized.

5.4.5 Knowledge transfer between databases

Fig. 5.8 shows the result of testing models trained on one database (CHB-MIT) and utilizing (testing) them on a different database (Repomse). More specifically, the performance of five

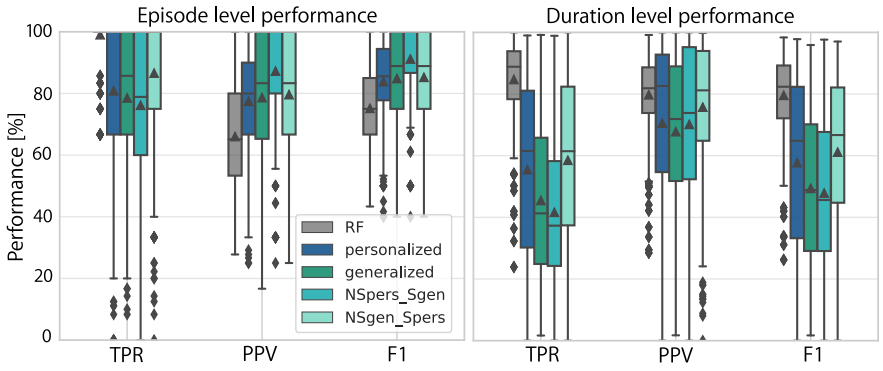

Figure 5.7: Performance of 4 types of models: personalized, generalized and hybrid (*NSgen-Spers* and *NSpers-Sgen*). Boxplots represent performance over all subjects. Sensitivity (TPR), precision(PPV) and F1 score both for performance on the level of episodes and on the whole duration level are shown.

different models is shown: personalized models and generalized models trained on Repomse, followed by three models trained on CHB-MIT database: generalized and hybrid (*NSgen-Spers* and *NSpers-Sgen*). Boxplots represent the performance distribution over all subjects. First, a clear drop in F1 score (for episode and duration) when using a generalized model from the CHB-MIT dataset is visible. This is due to much worse seizure detection as measured by significantly lower sensitivity (TPR) both on episode and duration levels. The *NSpers-Sgen* approach doesn't help to improve performance, but the *NSgen-Spers* does help. Indeed, personalizing seizure models improves sensitivity and also the final F1 score (both for episodes and duration), reaching and even exceeding the levels of the generalized model from the same Repomse database.

Fig. 5.9 shows the opposite transfer of models. Models trained on Repomse dataset with many subjects are tested on the CHB-MIT dataset. Here *NSpers-Sgen* models performed better, as in Repomse there is a very small amount of non-seizure data, and thus it doesn't generalize well for a dataset with much more non-seizure data. On the other side, Repomse contained more subjects and thus seizure data seems to be general enough to use generalized models.

5.5 Discussion

By analyzing inter-patient similarities, we show that non-seizure models are more similar than seizure models. This makes sense, as seizures usually exhibit specific individual patterns. Still, seizures to non-seizures are the least similar, enabling classification. For long-term, unbalanced epilepsy recordings, these trends would probably be even stronger, especially regarding non-seizure models. Further, this opens the possibility of checking similarities between individual patients and potentially grouping them into groups of similar seizure (or non-seizure) patterns.

Furthermore, we studied the creation of generalized models by adding personalized models

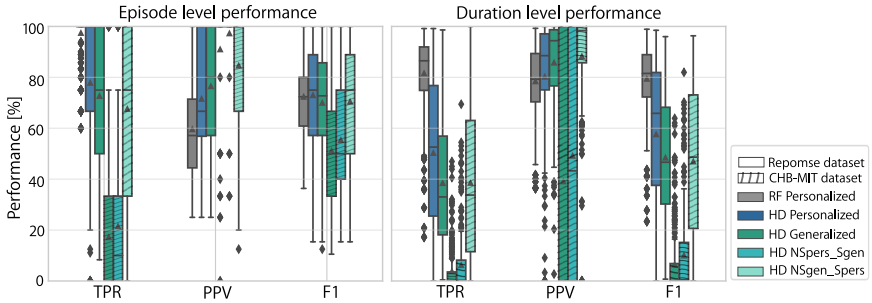

Figure 5.8: Knowledge transfer between two databases is tested on the Repomse dataset. Performance of 5 types of models is shown: personalized, generalized Repomse (models from Repomse data), generalized CHB-MIT, and hybrid from CHB-MIT (*NSgen-Spers* and *NSpers-Sgen*).

one-by-one in three different ways. It is shown that simple averaging of personalized models leads to the most similar vectors to the summed personalized ones, but also leads to low class separability. Approaches where vectors are weighted before being added, and also the idea of subtracting opposite class, leads to more separable generalized vectors. Thus, we recommend using one of those approaches for creating generalized model vectors. Iterative passing through subjects doesn't lead to better-generalized models. This is most likely due to the fact that we used 286 subjects, and generalized models were very stable, so passing again through individual models didn't change generalized models anymore. Assessment of the evolution of generalized models as, one-by-one, more individual subject models were added, lead to the conclusion that at least 40 individual subjects are needed to get stable generalized model vectors. We have to consider that different databases with unbalanced datasets result in different minimal number of subjects. For example, if data is long-term (such as, for example, in CHB-MIT epilepsy database), maybe non-seizure vectors will stabilize much sooner, whereas, for stable seizure models, we will need many more subjects.

Observing the performance of personalized and generalized models led to the conclusion that there is an obvious stratification between subjects. This could be due to not enough seizures or very different seizures in some patients, which makes personalized models not precise enough. It can also be that some subjects have very specific seizures for which generalized seizure models don't perform well. Even if utilizing generalized models for all subjects would be the most practical, generalized models do not perform well for all subjects. In Fig. 5.6, we can see that when caring about sample-by-sample prediction, personalized models always perform better. With a simple system where personalized models are built only for subjects where generalized models performed worse than some preset threshold, it is possible to achieve even better performance than with only generalized (or even only personalized models). Specifically, for the Repomse dataset, it is possible to maintain generalized models for up to 80% of patients without loss of performance. This is probably dataset-specific, and numbers might be significantly different if long-term datasets are used.

Finally, we tested the detection performance of hybrid models, containing one generalized

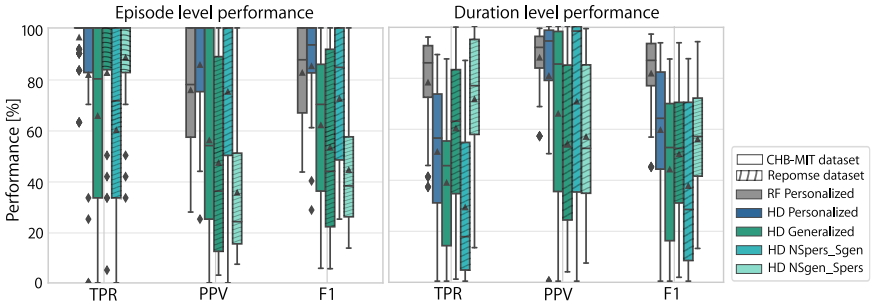

Figure 5.9: Knowledge transfer between two databases is tested on the CHB-MIT dataset. Performance of 5 types of models is shown: personalized, generalized CHB-MIT (models from CHB-MIT data), generalized Repomse, and hybrid from Repomse (*NSgen-Spers* and *NSpers-Sgen*).

and one personalized model vector (*NSgen-Spers* and *NSpers-Sgen*), and show that, indeed, they can improve overall seizure detection even more. For example, if we require high episode performance, the *NSpers-Sgen* model gives the best overall performance by decreasing the number of false positive seizure predictions. In contrast, if we care that the entire duration of the seizure is correctly classified, the (*NSgen-Spers* model performs better by keeping high sensitivity. We tested the possibility of keeping both personalized and generalized models for each subject and giving a prediction of a more certain model, but this led to giving a prediction of the personalized model in almost all the cases, thus not being particularly interesting.

When comparing the possibility of transferring knowledge between databases, interesting results were achieved. Transferring knowledge from the long-term CHB-MIT database to a shorter (but with more subjects) Repomse database showed that it was not good enough to use just a general model from CHB-MIT. Namely, seizure models from CHB-MIT showed low sensitivity (TPR) and F1 scores on the Repomse database. However, personalizing seizure models using Repomse data improved predictions significantly, leading to improved results compared to generalized (or even personalized) Repomse models. This means that for epilepsy detection on the Repomse dataset, it is possible to reuse non-seizure models from the CHB-MIT database and only retrain seizure models. When transferring knowledge from Repomse to CHB-MIT, an opposite effect was noticed. Seizures from Repomse were general enough, but non-seizure models were not general enough to use on a dataset that contained much more non-seizure data. Thus, to avoid full training on CHB-MIT, seizure models from Repomse can be used. All these results show that having a big long-term database with many subjects and building generalized models trained on that database could be potentially reusable for new unseen patients (without having to retrain models). If needed, as in *NSgen-Spers* and *NSpers-Sgen* hybrid approaches, only one class could be retrained, saving a lot of training time.

5.6 Conclusion

In this chapter, we have demonstrated how HD computing, and the way its models are built and stored, can be used to further understand, compare, and create more complex machine learning models. These possibilities are not feasible with other state-of-the-art models, such as random forests or neural networks.

We compared the inter-subject similarity of seizure and non-seizure epilepsy models and then studied the process of creation of generalized models from personalized ones, which is an essential part of distributed wearable applications. We tested different approaches to creating generalized models and showed that blindly aggregating all subjects can be improved by measuring the novelty of each subject before adding his personal model to the general model. Moreover, we estimated that at least 40 subjects are needed to be able to create a stable generalized model. However, this can be dataset specific.

We proposed a novel hybrid approach to create models that consist partially of personalized and partially of generalized models. This is a unique possibility that hyperdimensional computing allows, and we showed that these models result in improved epilepsy detection performance. More specifically, this enables us to overcome the limitations of some datasets, such as having a very small amount of seizure or non-seizure recordings.

Similarly, we use hybrid models to test knowledge transfer between models of different epilepsy databases and show that, indeed, there is the possibility to reuse models from one dataset on a different dataset. However, it is important to note that the limitations of datasets have to be taken into account in this process.

Finally, all those examples could be very interesting not only from an engineering perspective to create better models for wearables but also from a neurological perspective to understand individual epilepsy patterns better.

Limitations

The Repomse dataset is very interesting due to the high number of subjects available, which enabled us to study properly generalized models. As we have shown, we indeed needed more than 40 subjects to achieve stable generalized models. This is more than what is available in other public datasets (i.e., 24 subjects in CHB-MIT or 18 in SWEC-ETHZ [269]). However, it is a short-term database, meaning that performances might be overestimated with respect to what they would be on a long-term database. Thus, we studied knowledge transfer between slightly bigger data subset from the CHB-MIT database, using 10 times more non-seizure data. Ideally, in the future, a similar analysis should also be performed on a long-term database with a similarly large number of subjects.

From a neurological perspective, as the CHB-MIT database contains pediatric patients and the Repomse adult population, mixing the types of patients might not be that compelling. Furthermore, databases contain a broad range of seizure types but, unfortunately, no labels.

Mixing different seizure types adds a certain level of complexity. Here we were interested in the performance possible even in this limited scenario, but in the future, repeating this analysis separately for different types of seizure would be more interesting from a medical perspective.

6 Interpretability

6.1 Importance of interpretability

Even though AI-driven systems have been shown to outperform humans in certain analytical tasks, especially in the medical domain, the lack of explainability continues to spark criticism, making it one of the most debated topics when it comes to the application of artificial intelligence in healthcare [1]. The issues with explaining the outcomes of ML algorithms are relevant to medical practitioners, policymakers in public health, and end users, usually patients. They not only want an accurate model but also to understand how the model works, what recommendation has been made by the model, and why. These aspects have been emphasized by a number of recent studies [369], [370].

As a result, the area of interpretability and explainability of ML is gaining significant research momentum. Today, there is already a wide collection of interpretable ML methods and post-hoc methods for the interpretation of ML models, which were systematically compared and categorized in [371]–[373]. For example, in [373], authors created nice visualization of the taxonomy of interpretable ML approaches as shown in Fig. 6.1. Furthermore, the definition of interpretability is quite broad, and interpretable ML often comes under different names, namely, explainable ML, intelligible ML, or transparent ML. But the common part is the extraction of relevant knowledge from an ML model concerning relationships either contained in data or learned by the model. The optimal form of presenting that knowledge will often depend on the end user, i.e., the needs of an engineer designing the models will differ from those of clinicians.

Thus, an explanation can come in different forms: decision trees, decision rules, feature importance, saliency masks, sensitivity analysis, or more [374] Some authors argue that models with inherent explainability will, in general, be more accurate in their explanations than post-hoc methods that measure only approximate explainability [375]. However, other authors argue that still, most inherently explainable models are also inferior in terms of task performance [376]. Thus, here we try to explore the inherent explainability that the HD approach has, and demonstrate it through feature and channel importance and selection

process. To the best of our knowledge, the topic of interpretability in HD computing has not yet been researched.

In the end, visual analytics has become an important aspect of ML interpretability and even individual research field [377]–[379]. Thus, we showcase possibilities for visualization of contributions of individual features or channels to final predictions in time.

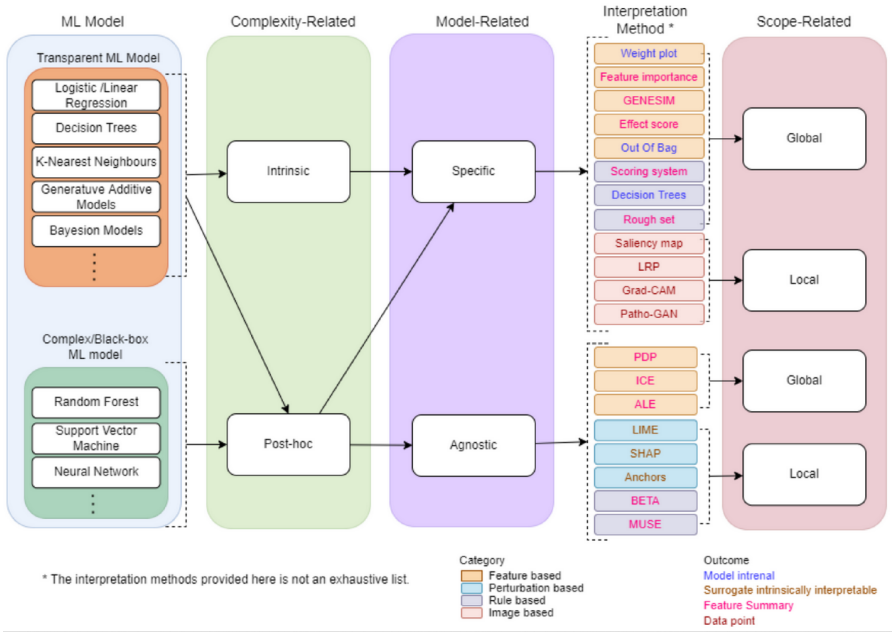

Figure 6.1: Taxonomy of interpretable ML approaches proposed by [373]. They can be classified based on complexity (intrinsic/post-hoc), model (specific/agnostic), and scope (global/local).

6.2 Feature comparison and selection

6.2.1 Motivation

Encoding is one of the crucial steps in HD computing workflow. How data is encoded was studied in Sec. 3.4. But, encoding can also enable analyzing performance, correlations, and learning capabilities of individual features. In [240], authors explored a similar idea using several individual HD classifiers for three different features. Analyzing various aspects of individual features is a crucial step for a) feature selection, as will be demonstrated in this section, and b) for further interpretability of model decisions, as will be discussed in section 6.4. Similarly, in [380], authors compared the importance of individual EEG channels using Fisher score and class separability. Feature (and channel) selection is crucial in designing wearable applications as it helps remove noisy and non-informative features (and channels) while also directly leading to more lightweight models. However, so far, there has been very little discussion and proposals for feature and channel selection as part of the HD computing workflow.

Thus, in this chapter, we focus on a designing methodology for feature selection inherent to HD computing workflow. More specifically, we demonstrate it by choosing an adequate encoding. Despite being a pretty straightforward approach, we hope that it can foster research in the ML community for further development in this direction. Broadly speaking, from the clinical perspective, tools and models that can improve the explainability of models, pinpoint the best features, or visualized decisions in time can lead not only to better service to the patients, but also to be decision support for the doctors and caregivers.

The contributions of this section are:

- We proposed an encoding approach that enables the quantification of several feature quality measures based on HD computing.
- We utilize this approach to further perform feature selection in several different ways, and show that, indeed, it is possible to decrease the number of features by using information from the encoded data.
- Based on our knowledge, this is the first attempt to have an in-build way for feature selection as part of the HD computing model.

6.2.2 Feature selection methodology

In section 3.4, we introduced an approach called $FeatAppend$, where feature information was not encoded using binding information but rather appended. In this approach, it is known which part of the final encoded vector comes from each feature, which enables analysis per individual feature. Thus, we can define and measure several metrics for each feature:

- **Prediction**: Using only $d = D/nF$ bits (where D is the final hypervector dimension, and nF is the number of features) corresponding to the feature of interest, we can determine

the prediction for each sample using only that feature. A decision is made in the same way as when the whole vector is used; the label of the most similar class vector is given. As we use binary HD vectors, hamming distance is used as our similarity metric measure.

- **Feature certainty**: For each time moment, we can also quantify the certainty of each feature's label based on distances from both class vectors. The certainty is calculated as the difference between distances from two classes, divided by the average absolute distance for all features in the same time moment, as in Eq.6.1.

$$C_f = \frac{|dist_{Sf} - dist_{NSf}|}{\frac{1}{nF}\sum_{i=1}^{nF}(|dist_{Si} - dist_{NSi}|)} \tag{6.1}$$

- **Correlation**: Based on the predictions of each feature in time, it is possible to measure the correlation between predictions of different features and take it into account later for feature selection.
- **Class separability**: For binary HD vectors, separability (S_f) is measured as the relative hamming distance between class vectors (HD_S, HD_{Ns}) when using only the bits of the corresponding feature. This measurement is obtained using Eq.6.2. Based on the separability measure, it is also possible to estimate the usefulness of each feature.

$$S_f = hamm\left(HD_S\left[\frac{f*D}{nF} : \frac{(f+1)*D}{nF}\right], HD_{NS}\left[\frac{f*D}{nF} : \frac{(f+1)*D}{nF}\right]\right) \tag{6.2}$$

Metrics measured per feature on a training set can then be used to perform feature selection, as shown in Fig. 6.2. In this paper, we demonstrate three ways to do it. Each approach starts by ordering features based on specific quality measures:

- **Feature performance**: Based on the predictions for each sample and the true labels, we can measure the performance of each feature. The exact performance metric of choice can depend on the application.
- **Feature confidence**: Based on the certainty values and predictions per feature, as well as true labels per time moment, from the training set, features can be ordered based on the highest confidence. Confidence per feature is calculated as how much more certain that feature is during correct predictions ($C_f|TP$) versus wrong predictions ($C_f|FP$) and is calculated by Eq.6.3.

$$Conf_f = \frac{\overline{C_f|TP} - \overline{C_f|FP}}{\overline{C_f|FP}} \tag{6.3}$$

- **Feature performance and correlation**: If selection is made based only on the feature performance, it might lead to selecting features that are performing well but are highly correlated and thus probably redundant. In this approach, we select features one by one by evaluating in each step how the performance changes when adding one of the not yet used features. This is a slower process (and computationally more complex) than sorting features based on different direct quality metrics, as in the previous two

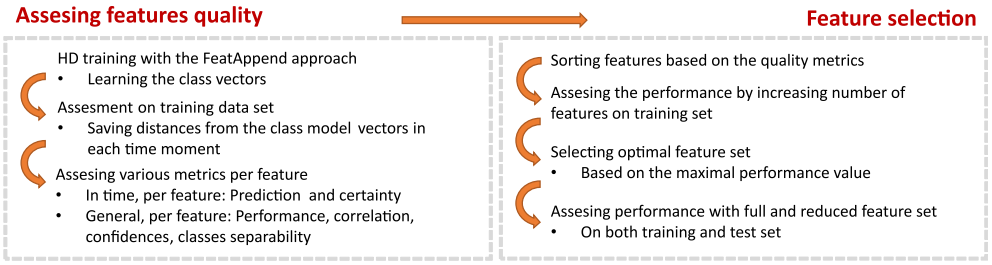

Figure 6.2: Feature selection workflow with detailed steps.

approaches. Nonetheless, it leads to a more elaborate order of features, which also includes the novelty each feature brings.

After features are ordered, the performance on the train and test set is assessed when adding one-by-one features until all features are included. Prediction when using n features is given by summing up distances from seizure and non-seizure model vectors of individual features as in Eq. 6.4. If the final distance is positive, the prediction (*Vote*) is a seizure, otherwise non-seizure. From the performance curve of the training set, the optimal number of features is chosen as the number of features giving maximal performance (or the smallest number of features without loss in performance when compared to using all features). In the end, as shown in Fig. 6.2, performance on the test set is measured for the chosen, reduced set of features. Then, we compare it with the case without feature selection.

$$Vote = sign\left[\sum_{i=1}^{nF}\left(dist_{NSi} - dist_{Si}\right)\right] \tag{6.4}$$

6.2.3 Experimental setup

The experimental setup used for feature selection is identical to the one used in the previous exploration of optimal encoding approaches for spatio-temporal data. It is in detail described in Sec. 3.4.3.

6.2.4 Results

Fig. 6.3 shows a comparison of the features based on measures extracted using *FeatAppend* approach. More specifically, the separability of vectors for the two classes are shown for each feature. Next, the average confidence of each feature and the performance (F1 score for episodes, duration, and their gmean) when using only individual features are shown. In the subplot of performance, the horizontal line shows the performance achieved when using all the features. It is visible that no single feature reaches the performance of all features, but some of them get quite close to it (i.e., mean amplitude, line length, total power, and power of delta and theta). In the bottom row, the correlation between confidence and performance, separability and performance, and separability and confidence are plotted, showing high

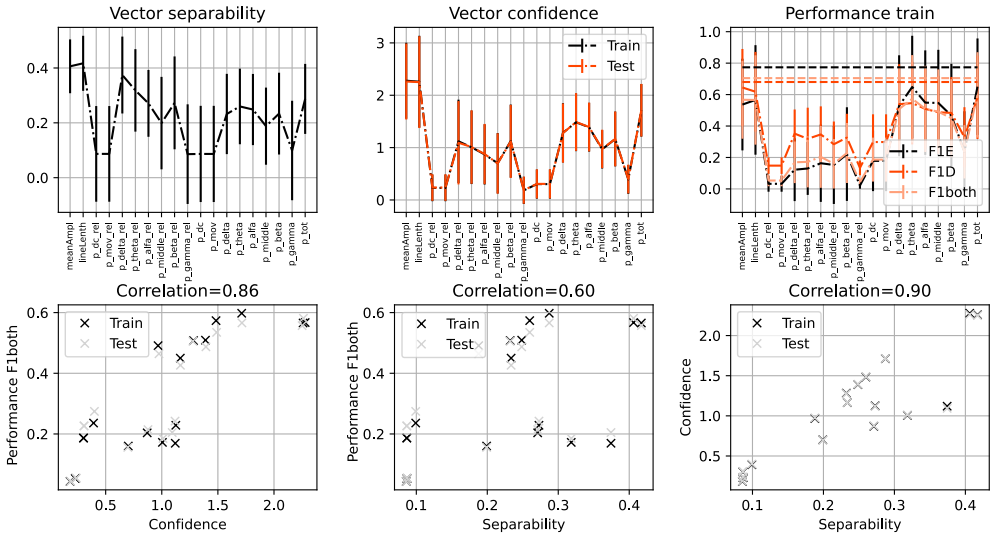

Figure 6.3: Comparison of features based on *FeatAppend* approach. The averaged values over all subjects are shown.

correlation values. This confirms that the *FeatAppend* approach can indeed be used to investigate different properties of individual features.

Fig. 6.4 shows the performance when adding features one-by-one for each of the three methods described in Sec. 6.2.2. The difference between approaches relies on how features are ordered. In the first row of Fig. 6.4, features are ordered based on their performance metric (in this case, their F1DE performance), and in the second row, based on average feature confidences given by Eq.6.3. Finally, in the third row, features are added one by one choosing the best feature to increase performance when added to previously chosen features. This approach can be referred to as more optimal order and choice of features, as it takes into account the correlation between features. In this case, performance increases with respect to using the first two approaches.

The last column shows the boxplots of feature orders for each feature selection method for all subjects. The smaller the ranking number, the sooner that feature was chosen in the incremental feature selection, on average, for all subjects. What can be noticed is that the ranking of features in the first two approaches when using feature performance or feature confidence solely is similar to the results given by the discriminative power analysis shown in Fig. 3.13. In the last case of more optimal feature selection, feature order is different due to feature correlations that were taken into account.

Fig. 6.5 shows the results of optimal feature selection for every subject. The first graph shows the chosen number of features per subject and the average for all subjects, which is shown with a horizontal line. The following two graphs show the performance improvement (or decrease) for every subject (and average with the horizontal line), i.e., F1 for episodes and the geometric mean of F1 for episodes and duration, for the training and testing set. In the title of the figures, we provide the average performance for all subjects on the test set. Significant

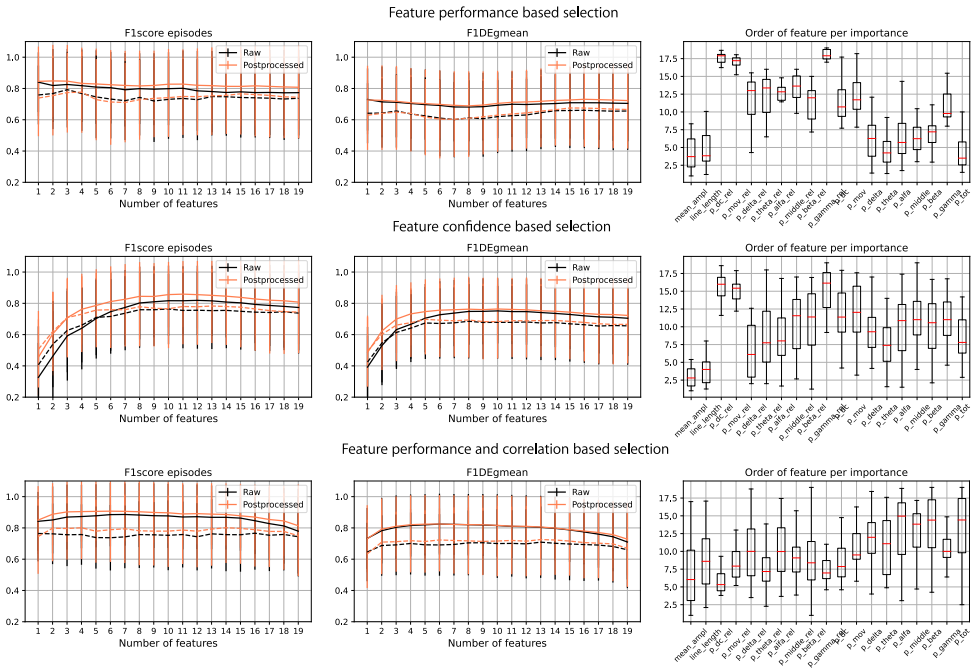

Figure 6.4: Performance evolution by incrementally adding a one-by-one new feature for three approaches: based on feature performance, feature confidence, and both feature performance and correlation.

variability exists between the number of features chosen per subject, ranging from 1 to 10 features, with an average of 5.8 or 30% of features.

Table 6.1 shows more detailed results. More precisely, results for all three feature selection methods are given. For each of the methods, we show results when we optimized the F1 score only for episodes (F1E), or when the F1 score for seizure duration is also taken into account (F1DE). Further, in the table, average F1 and F1DE performances for both train and test are shown before and after label post-processing. What can be noticed is that when optimizing only F1E, fewer features are needed, but it usually leads to a smaller performance increase for F1DE. When F1DE is optimized, this leads to a significant performance increase both for F1DE and F1E but at the price of a slightly higher number of features chosen. In general, the performance increase is smaller on the test set than on the train set, which is reasonable as the optimal number of features was chosen based on the train set without knowledge about the test set. Yet, the performance increase is not negligible, ranging up to 7% for the test set.

When comparing the three different feature selection approaches, there are slight differences in the optimal number of features chosen and the performance gained. Feature selection based only on performance or confidence leads to a slightly higher number of features but not a higher performance increase. This is due to information about feature correlations that are missing.

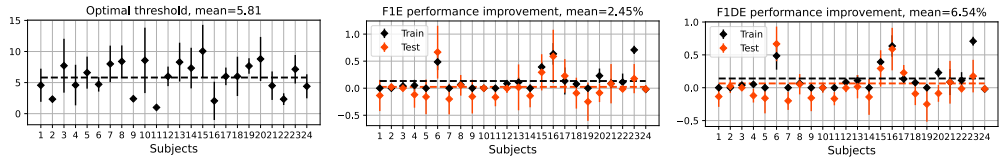

Figure 6.5: Optimal number of features and performance after feature selection. An example is shown for feature selection using both feature performances and correlations. The horizontal lines represent the average for all subjects.

Table 6.1: Optimal number of features and performance change for different feature selection approaches.

Feat. selection approach	**Perf. used**	**Nr. Feat.**	**Train Raw**		**Train Post**		**Test Raw**		**Test Post**	
			F1E	**F1DE**	**F1E**	**F1DE**	**F1E**	**F1DE**	**F1E**	**F1DE**
Feature performance	F1E	2.84	12.29	4.63	6.75	1.43	4.78	-1.40	-0.27	-4.53
	F1DE	6.99	9.41	7.49	6.60	5.39	4.94	3.69	3.64	1.90
Feature confidence	F1E	5.98	10.92	8.55	6.59	5.56	4.00	3.75	2.90	2.28
	F1DE	6.62	9.44	10.26	6.60	7.81	3.56	6.03	3.43	4.82
Feat perf and corr	F1E	2.65	14.61	8.91	9.35	5.45	5.84	1.88	1.79	-1.06
	F1DE	5.81	13.29	14.07	9.44	10.84	2.45	6.54	5.27	6.96

6.2.5 Conclusion

By utilizing the *FeatAppend* approach, we present, a way to perform feature selection using HD computing. An incremental feature approach was tested with three different methods to determine the order of features to be added. All approaches led to a significant reduction of features while keeping or even significantly improving the performance compared to using all features. In the future, approaches using feature elimination could be similarly tested.

FeatAppend is interesting not only from a feature selection perspective but also from the clinical perspective of a deeper understanding of various features and their properties. For example, we investigated several measures per feature: performance, probabilities of decisions, confidences, correlation, and separability of classes, which can all lead to knowledge discovery related to the usefulness of features. This approach can be adapted to *ChannelAppend* and be used in an analogous way for channel comparison, channel selection, and potentially seizure localization, which will be shown in the next chapter.

6.3 Channel selection

6.3.1 Motivation

Studying predictions on the level of individual channels has two main opportunities: 1) it can enable localization of the seizure and potentially even track seizure spread over the brain surface, and 2) it can be used for reduction in the number of channels used in ML models for seizure detection/prediction.

Localization of the seizure onset is particularly important in the presurgical assessment. Before there was (I)EEG, clinical seizure localization relied mainly on postmortem studies. Then, (I)EEG enabled visual inspection, detecting the ictal onset and observing the seizure spread. Recently, researchers started developing deep learning approaches for automatic detection of seizure onset [381], [382].

Similarly, different approaches have been tested for channel reduction to make models more viable for wearable implementations. However, many approaches are based on a) calculating different metrics for each channel and selecting a pre-defined number of best-ranked channels [383]–[386], or b) testing several channel combinations and choosing the best one [387]–[389]. Only a couple of works developed a completely automated approach that finds the optimal number of channels for each subject [390].

Furthermore, to the best of our knowledge, HD computing has not been used for channel selection. In recent work [239], authors used LBP based HD model for the identification of ictogenic brain regions. However, the resolution they used was either on the level of cerebral lobes or hemispheres.

Thus contributions of this chapter are:

- We demonstrate a simple approach to how HD computing can be used for channel selection in a completely automated way. The approach is based on $FeatAppend$ encoding.
- For each subject chosen number is automatically determined. Individualized composition of optimal channels is also visualized in topo-plots in the shape of the head.
- We show that, on average, we can lower the number of channels from 18 to 5 without losing any performance on the test set. It has also shown that, on average, keeping all channels is not optimal from the performance perspective, either.

6.3.2 Experimental setup

The experimental setup used for studying the optimal choice of channels is identical to the one used in the previous channel for feature selection. In fact, it is explained in the exploration of optimal encoding approaches for spatio-temporal data in Sec. 3.4.3.

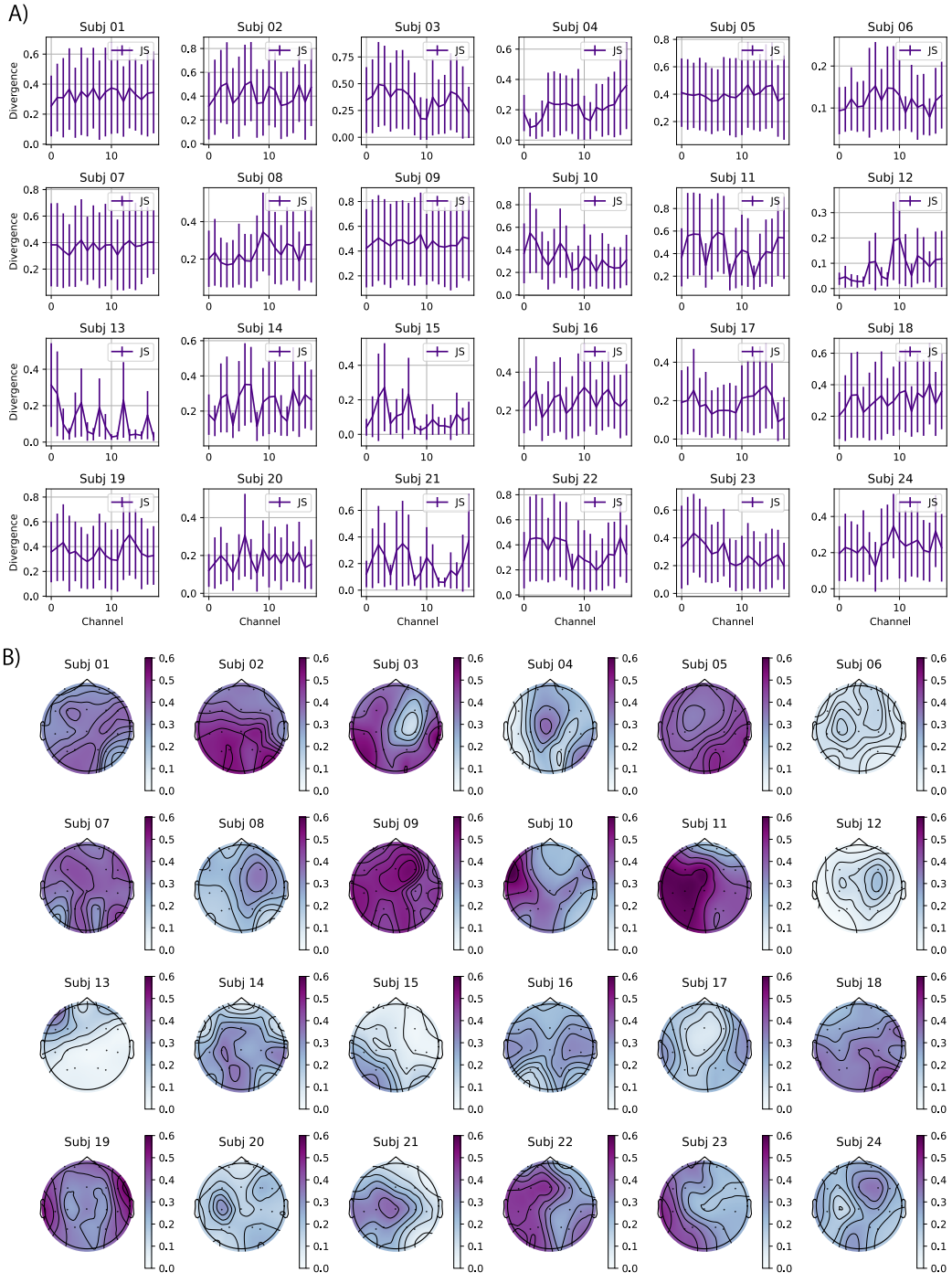

Figure 6.6: Jensen-Shannon divergence per channels and per individual subjects. The average and standard deviation for all features is shown.

6.3.3 Results

Comparing individual channels

The simplest approach to compare channels is to measure the Kulbeck-Leibler (KL) divergence of feature values during the seizure and non-seizure period on each specific channel. But since the maximal value of KL divergence is not bounded, we instead use a normalized version of Jensen-Shannon (JS) divergence and calculate for each individual feature on that channel. Fig. 6.6A) shows the average over all features for each channel and all subjects. In Fig. 6.6B), mean values were shown on a topo-plot representing all recorded EEG channels visually. For some subjects, it is easy to notice more discriminative channels (for example, for subjects 03, 04, 10, 11, 13, 21, or 13), whereas, for some subjects, there does not seem to be a lot of difference in the usefulness of channels. For example, subjects 02, 05, 09, and 11 have quite discriminative channels, while subjects 06, 12, 15, 17, or 20 have very poorly discriminative channels.

Selection of optimal channels

As explained in the previous chapter, $FeatAppend$ approach enables analyzing performance per individual channel as well as their correlations. This information can be then used to perform incremental channel selection by starting with the best-performing channel, and then adding one by one channel, analyzing performance with the added channel. This approach of choosing the optimal next channel based on its performance but also a correlation with previously chosen channels is utilized here as well. In Fig. 6.7 detailed examples for the first two subjects of CHB-MIT database are shown.

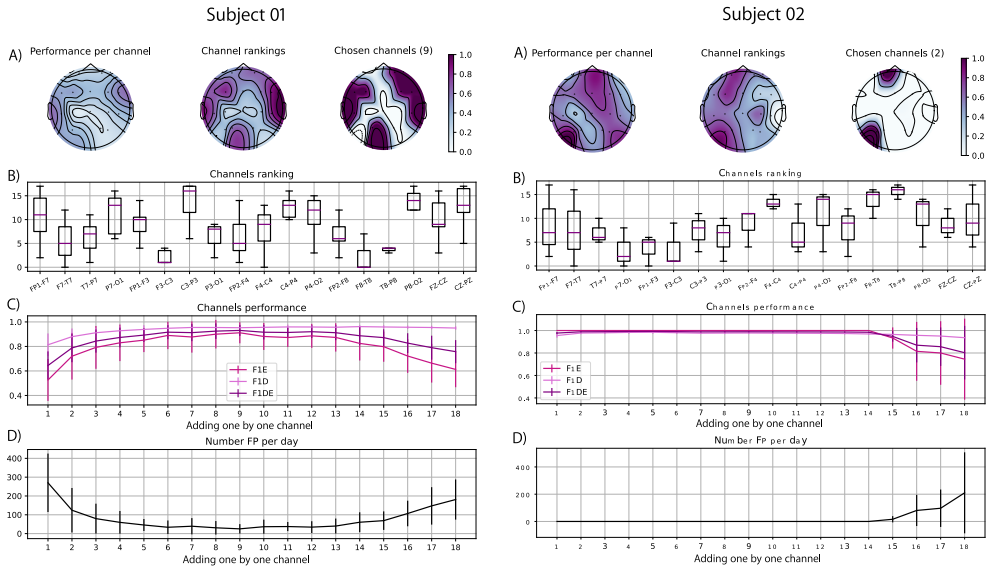

Figure 6.7: Example of channel selection for two subjects of CHB-MIT dataset (subjects 01 and 02). Performances per feature, optimal ordering of features, performance as the increasing number of features, and finally chosen set of features are shown.

More specifically, on the first topo-plot on figures *A*) we show performance (geometric mean

of F1 score for episode and duration) per individual channel. Next, taking into account predictions of each channel (that can be calculated using hamming distance and bits of vectors corresponding to channels of interest) we calculated the optimal order in which channels will be included to bring the best performance improvement with each next channel added. This is called the ranking of the channels and is shown in figures *B*) as well as the second topo-plot of figures *A*). In figures *B*), the smaller the number the earlier the channel was chosen. Boxplot represents distribution over all recording files of that subject, where in each file was one seizure. In figures *A*), the darker the color, the more important the channel was. Next, in figures *C*) we visualized performance as a one-by-one channel was added. We show performance for both F1 scores for episodes, duration, and their gmean. It can be easily visible that improvement can be achieved by adding more channels, but that also using all channels does not necessarily means the best performance. A similar trend is visible in figures *D*) where a number of false positives per day is shown. Finally, the optimal number of channels chosen that maximize performance and minimize the number of them is shown on the third topo-plot of figures *A*). We marked the number of chosen features in the subtitle of the figure.

We can notice that usually, the most performant features will be in the selected set at the end. But, it can also be noticed that in some cases, channels that might not have had high performance and are often far from performant channels are chosen in the end. This can be seen, for example, for subject 01 in Fig. 6.7, where parieto-occipital channels P3-O1 and Pz-Oz were chosen too. In the end, the performance per channel of each subject and, finally chosen set of channels for every subject is shown in Fig. 6.8. Similarly, as in JS divergence analysis, some subjects have clearly visible channels that, even as individual channels achieve quite high-performance values (e.g., subjects 02, 03, 07, 09, 10, 11, 19, 21, 22, and 24). This subject usually results in a smaller selection of channels, where sometimes even not all well performing channels are needed (e.g., subject 09 or 11). On the other hand, some subjects have very low performance for almost all channels (e.g., subjects 01, 04, 13, 18, or 23). These subjects consequently require more channels to be chosen to reach optimal performance (e.g., subjects 01, 13, 14, 17, 18, or 20).

Finally, Fig. 6.9 shows the average performance over all subjects by adding channels one-by-one. The order of channels is chosen on a train set, and performance is also measured on a test set. The general trend of increasing performance as more channels are added is visible, but also that after a certain amount of channels performance drops. This is due to the fact that in many subjects, seizures are focalized, and channels far from the seizure might not be able to discriminate between seizure and regular EEG, making classification harder when included. An optimal number of channels on average is around 5, which can also be seen on the first graph in Fig. 6.10. More specifically, the optimal number of channels per subject is also shown here, as well as performance improvement on both the train and test sets. Overall, channel selection leads to high performance improvement on the train set but not a particular improvement on the test set. However, there is no performance degradation.

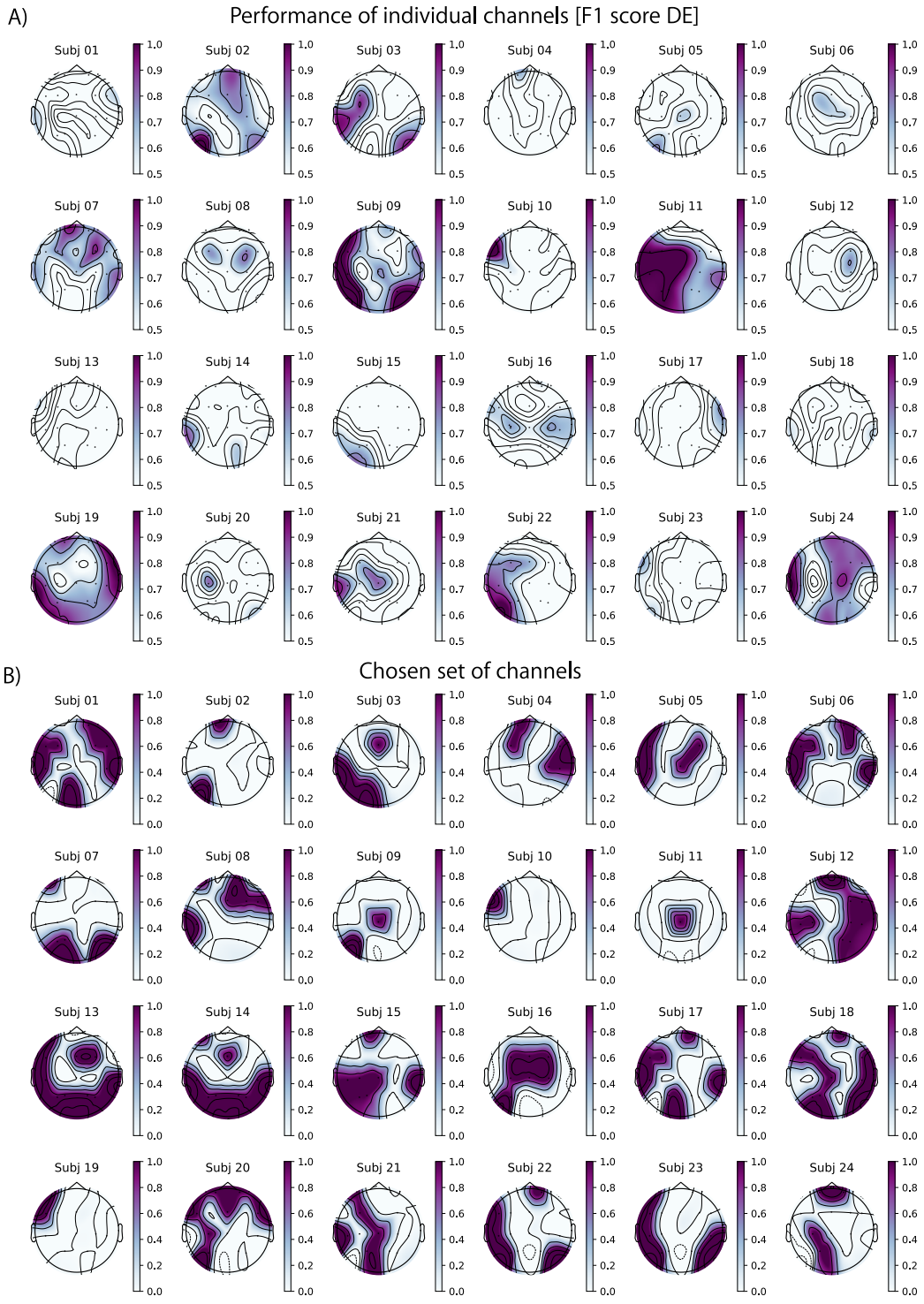

Figure 6.8: A) Performance (geometric mean of F1 score for episode and duration) per individual feature is shown for every subject. B) Finally chosen optimal set of channels is visualized for every subject.

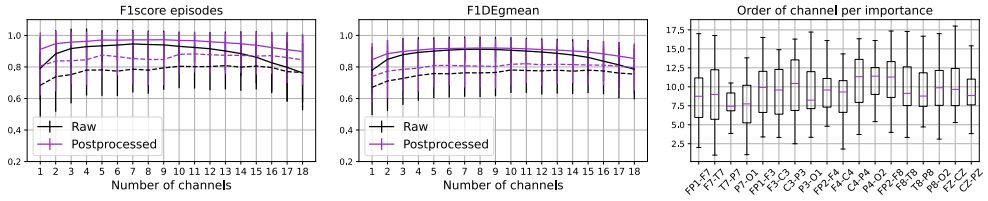

Figure 6.9: Average performance over all subjects as adding one by one channel. The order of channels is chosen on a train set, and performance is also measured on a test set.

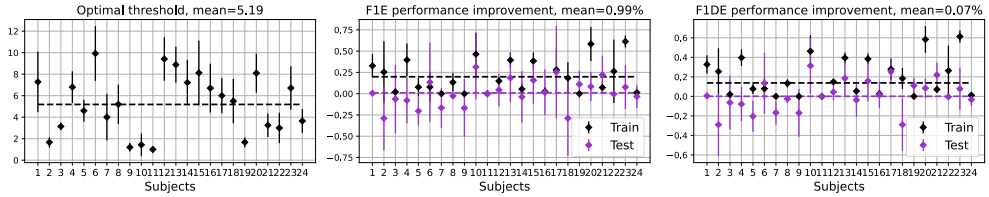

Figure 6.10: Number of channels chosen per feature and performance increase (drop) for each subject when compared to the case with all channels. Horizontal lines represent the average over all subjects.

6.3.4 Conclusion

By utilizing the *FeatAppend* approach, we present a way to perform channel selection using HD computing. An incremental channel approach was tested. We have shown that, indeed, it is possible to reduce the number of channels from 18 to 5, on average. Important to note is that this approach completely automatically decides the optimal number of features for each subject. We also visualized in a simple way how decisions were made and what were final chosen channels in the form of EEG scalp topo-plot.

FeatAppend is a simple and interesting approach not only from a channel selection perspective but also for potential future seizure localization. Moreover, in the next section, we will show how it can be also used for interpretable visualization of predictions of the model.

6.4 Interpretability in time

6.4.1 Motivation

Finally,we want to demonstrate the ability to visualize, in time, predictions per individual channel or feature. This is a unique opportunity that specific types of HD encodings can enable. Namely, suppose the encoding is designed in such a way that not all features/channels contribute to all bits or contribute with different weights. In that case, it is possible to backtrace how much each channel or feature is contributing to the prediction of seizure or non-seizure and with which confidence. Here we demonstrate this using $FeatAppend$ approach, but more elaborated approaches should be tested too.

Understanding the contributions of each feature and channel in time is extremely valuable to observe reasons for false predictions. Suppose all features and channels are very consistent in predicting seizure when it is not officially labeled. In that case, it can draw attention to such areas, requiring their further inspection in detail, and can potentially lead to either detecting missed seizures or potential noise types that need to be taken into account. Moreover, observing predictions on the level of channels and topological distributions on the scalp can also enable monitoring of the origin and spread of seizures through the scalp and EEG channels. Moreover, slight disagreements in starting and ending of the seizures can be noticed.

Most of the 'explainable' ML models do not enable this resolution of predictions. Usually, they are able to point out more important channels and more important features, but not with such precise time resolution. In other cases where time resolution is available, such as with decision rules, it is often not easy to visualize it.

So far, we have not encountered an HD computing paper proposing such an approach, and thus we want to draw attention to the potential HD computing has in the area of interpretability and visual analytics.

6.4.2 Experimental setup

The experimental setup used is the same as the one used in the previous two chapters. It is explained in the exploration of optimal encoding approaches for spatio-temporal data in Sec. 3.4.3.

6.4.3 Results

Time predictions per channels

In Fig. 6.11, we showcase predictions in time per individual channel for several subjects. For each channel, a prediction is made using all available features. We color-marked if the channel predicts seizure (red) or non-seizure (blue). Moreover, with the intensity of the color, confidence in a given prediction is shown. On the top row, the final prediction using all

channels is shown. Finally, with the red line, true labels with the onset and end of each seizure are shown.

Several interesting things can be noticed in the plots. For example, individual channels often have lots of false positives, but accumulating all channels together filters out individual channel false predictions (e.g., see Subj 03, 08, 11). In our case, simply using all bits prediction instead of individual ones for each channel results in, effectively, majority voting from all channels. However, in practice also, more strict or lenient voting could be put in place (e.g., that more than 75% channels need to vote for a seizure to finally predict a seizure). Next, it can be clearly seen that some channels have quite a lot of false positives (e.g., channels $P8-O2, T8-P8, P7-O1$ and $T7-P7$ for Subj08, or $FP2-F8$ and $FP2-F4$ for Subj11). This would suggest excluding these channels for that individual subjects.

When looking at predictions during seizures, sometimes discrepancies in the beginnings and ending of predicted seizures and labels are visible (e.g., subj 08 and 19). For example, it seems that for subject 08, most detected seizures are shorter than the given labels. This could be due to the hard decision by medical personnel for the true end of seizure because of similar post-ictal patterns. This also clarifies the need to observe and quantify performance not only on a sample-by-sample (duration) basis but also on the episode level. For subject 19, it is interesting to notice delays in the onset of seizures per different channels. It seems like a seizure was first detected in the frontal regions ($F7-T7, Fp1-F7, F8-T8, FP2-F8$) and last at the back of the head ($P3-O1, P4-O2$). This potentially signalizes the seizure origin and spread through the scalp, but it would have to be confirmed by a neurologist.

Time predictions per features

Using the same approach for channels, we repeated the analysis for visualizing predictions per individual feature. Examples for the same four subjects are shown in Fig. 6.12. Similarly, as above, for one feature, all channels are included in predictions of that feature. It is clearly visible that some features are less helpful for classification in some subjects (e.g., features p_gamma for subject 03, and p_dc, p_mov, p_gamma for subjects 08 and 19. In some cases, those features are only predicting non-seizure(subjects 03 and 08), or mostly seizures (subject 19).

Similarly, as in analysis per channels, differences in seizure duration when compared to true labels can be seen. For example, in subject 08, most of the features detect only the beginning of the seizures. In subject 19, on the other hand, most of the seizures detect onset much later when compared to the labeled onset. This analysis enables us to easily see when certain features detect false seizures and how this is distributed in time and with respect to true seizures.

Figure 6.11: Examples for four subjects of predictions per channel in time. Red color signifies that seizure is predicted, while blue signifies non-seizure. The intensity of the color is proportional to the confidence in the prediction. With a red line, true labels are marked.

Figure 6.12: Examples for four subjects of predictions per feature in time. Red color signifies that seizure is predicted, while blue signifies non-seizure. The intensity of the color is proportional to the confidence in the prediction. With a red line, true labels are marked.

6.4.4 Conclusion

In this section, we illustrated the potential of HD computing for the explainability of predictions by plotting predictions and confidences per individual feature and channel in time. We demonstrate that this approach, other than explaining individual decisions with high time resolution (rather than overall explanations through channel or features importance, as in previous chapters), can have other potential uses. For example, it can be used to detect missed seizures, noise types that cause false positives, or channels/features that might be redundant or too imprecise.

Further, due to the potentially very high temporal resolution, it can be used to compare labels with detected onsets and ends of seizures per individual channels or features, giving additional information about the seizure evolution in terms of signal characteristics and its location.

6.5 Conclusion

In this chapter, we tackle the 'black box' problem many machine learning models have and demonstrate that HD computing has high potential to not be a 'black box' but rather enable various approaches to increase understanding of how and what the model learns.

More specifically, designing the encoding of data (features calculated from different channels) in such a way that not all features/channels contribute to all bits of the hypervectors, or that they contribute with different weights, enables quantification of contributions of each feature/channel to the final prediction. More specifically, it is possible to backtrace predictions and confidences of each channel or feature in time. Here we demonstrate how this can be used for 1) feature selection, 2) channel selection, and 3) visualization of predictions in time per individual channels or features. We demonstrate this using a simple $FeatAppend$ encoding approach, but more elaborate approaches should be tested in the future as well.

In section 6.2, we demonstrate how features can be compared and further selected in the optimal order so that with the minimal amount of features, performance is improved for each subject individually. An incremental feature approach was tested with three different methods to determine the order of features to be added. All approaches led to a significant reduction of features while keeping or even significantly improving the performance compared to using all features. In the future, approaches using feature elimination could be similarly tested. We believe that this is interesting not only from a feature selection/reduction perspective but also from the clinical perspective of a deeper understanding of various features and their properties. For example, we investigated several measures per feature: performance, probabilities of decisions, confidences, correlation, and separability of classes, which can all lead to knowledge discovery related to the usefulness of features.

Then, in section 6.3, the same approach is used to perform channel selection. We have shown that, indeed, it is possible to reduce the number of channels from 18 to 5, on average. It is important to note that this approach completely automatically decides the optimal number of features for each subject. We also visualized in a simple way how decisions were made and what final chosen channels for each subject were in the form of EEG scalp topo-plots. This approach can also be interesting in the future for potential seizure localization.

Finally, in the last section 6.4, the explainability of predictions in time is demonstrated. More specifically, predictions per channel or per feature can be shown in an easily interpretable way. We demonstrate that this approach, other than explaining individual decisions with high time resolution (rather than overall explanations through channel or features importance, as in previous chapters), can have other potential uses. For example, it can be used to detect missed seizures, noise types that cause false positives, or channels/features that might be redundant or too imprecise. Further, due to the potentially very high temporal resolution, it can be used to compare labels with detected onsets and ends of seizures per individual channels or features, giving additional information about the seizure evolution in terms of signal characteristics and its location.

FeatAppend is a very simple approach, but it was sufficient to demonstrate the potential HD computing has for the explainability of its models. Based on our knowledge, this is the first time that this property of hyperdimensional computing has been tackled in the literature. We hope that this will inspire other researchers to design more elaborate and powerful encoding approaches that will be able to capture more complex relationships between features and channels and still keep interpretability possibilities.

Limitations

FeatAppend approach is a very simple approach that also has several limitations. First, the approach is not very scalable to cases with many features. Namely, the final hypervector dimension D increases as the number of features increases, or if D is kept fixed, then the number of bits per feature d reduces. Too low d can lead to poor performance per feature. 'ID-Level' encoding, where features would be bundled with channels and values, could, in theory, be used for decoding back contributions of each feature (or channel), using the 'release' operation. However, in practice, this is not so easy because, after many bindings, decoding results are only approximate values, which based on our experience, were not precise enough. Next, as each feature is appended, without additional operations, one can only give predictions using linear combinations of given features, which might be a limitation in some cases. We believe that more complex encoding would allow fixed vector size and more complex feature relationships and enable decoding back to contributions of individual features. For example, we tested variants of the approach proposed in a recent paper [391], but the system was too sensitive to an imbalance in the dataset. However, this is something to be worked on in the future.

Furthermore, we want to draw attention to the fact that performed channel selection is not necessarily the same as seizure localization. Namely, graphs like 6.8A) where performance per individual channel is plotted could look like localization of seizure, and indeed it would make sense that channels about ictogenic region are more discriminative and lead to better detection of seizures, but this does not necessarily have to be like that. In the CHB-MIT dataset, the best classifying channels or channels close to seizure onset are not marked, so we could not verify if this approach could also detect seizure onset. In general, datasets rarely have labeled channels above ictogenic region, or at least channels that neurologists find visually most useful for seizure detection. Such datasets would be very valuable and could enable us to compare how channel selection and seizure localization go together.

7 Conclusion and Future work

7.1 Conclusion

The main motivation of my thesis was twofold; one focusing on epilepsy detection and the other on exploring the advantages and limitations of hyperdimensional computing for biosignals monitoring.

The main motivation for focusing on epilepsy detection comes from the fact that a significant portion of the human population suffers from epilepsy; at the same time, a lot of effort is made by researchers in designing novel machine learning approaches, yet small wearable devices for non-intrusive long-term monitoring of epilepsy are still quite far from reality. First, we try to bring attention to important aspects like databases and experimental design. Namely, without continuous, long-term, and well-annotated databases that truly represent properties and distributions of real-life data, challenges such as unbalanced data, limited spatial recordings, and noise will remain difficult to properly address. This could lead to a significant reduction in performance and thus disappointment when algorithms are employed in real-life. We also defined several critical characteristics that make datasets suitable for certain model types or methodologies such as generalized/personalized training, seizure prediction/detection, or seizure classification.

The problem with current methodologies is that some methodological decisions are not appropriate for the development and optimization of algorithms when taking the above-mentioned characteristics of epilepsy into account. But furthermore, another challenge is that methodologies differentiate in many aspects in the literature, making a comparison of results extremely challenging. This makes drawing conclusions and directing the next research steps hard. Oftentimes, critical methodological choices are not mentioned in papers, and source code is not available. Furthermore, we characterized the influence of a broad range of methodological choices important for epilepsy detection systems and gave recommendations whenever possible. For example, performance metrics must reflect users' needs and be sufficiently sensitive to guide algorithm development. For this reason, we encourage the usage of both episode-based and duration-based performance metrics, which can together give a

more nuanced picture of algorithm performance. Finally, we also encourage researchers to utilize a more open approach when presenting and sharing their work to further increase the comparability and reproducibility of results.

We assessed that hyperdimensional computing has a set of interesting properties from both software and hardware perspectives, making it highly interesting for epilepsy as well as other biomedical applications. For example, its highly parallelizable implementation is important for efficient and low-power battery implementations. Further, online learning will be essential for high-performance continuous monitoring of challenging and highly personalized medical wearable devices. The basic HD computing approach is essentially an online form of learning where samples may be added one-by-one to the model without the need to store all previous data. Similarly, HD models in the form of HD vectors that can be compared and combined are ideal for distributed learning and knowledge transfer between individual wearable devices.

Thus, in this thesis, we focus on exploring HD computing for epilepsy monitoring as a representative case of various healthcare monitoring applications from biosignals. We identified several challenges that such applications pose and divided chapters based on them; 1) spatio-temporal data, 2) unbalanced and large datasets, 3) the personal nature of datasets, and 4) the need for explainability in the medical domain.

First, in chapter 3, we discuss how to optimally capture all spatial and temporal information in the process of encoding it to HD vectors. We systematically researched encoding different epilepsy-specific features and showed that, indeed, expert-knowledge-designed features could help in HD computing as well. Then we focused on designing one such feature that is inspired by neurologists' procedure in interpreting and labeling EEG data in clinical practice. The approximate-zero crossing feature captures both time and amplitude information, and has been shown to be highly discriminative and also easily interpretable. In the end, we proposed several ways to encode data while either accounting for or ignoring spatial distribution information in EEG channels. We have shown that encoding spatial data improves performance. We have compared several approaches to encode this information in terms of epilepsy detection as well as computational and memory complexity.

Next, in chapter 4, we study challenges resulting from highly unbalanced nature and also long-term recordings leading to large datasets. Different methodological choices, such as cross-validation and chosen performance metrics, are discussed. Then, we focused on enhancements to HD computing learning procedures that can improve performance on unbalanced datasets. More specifically, we proposed an approach that doesn't assume one HD vector per class, but rather allows more than one (thus named *multi-centroid*) per individual class. This is appropriate for epilepsy, as ictal patterns show high variability both between patients and also within patients, whereas inter-ictal data actually represents a whole spectrum of human activities that can have completely different EEG patterns. This is a semi-supervised approach that automatically determines an optimal number of centroids per class. Results showed that this can indeed significantly improve epilepsy detection performance. Next, we

compared the proposed *multi-centroid* approach with other literature proposals for improving HD learning, such as iterative and weighted (*'OnlineHD'*) learning. Finally, we tackled practical challenges when using HD computing on large datasets. Namely, there is still no common infrastructure in terms of code and libraries for training HD computing models. Thus we developed one such open-source library for training HD models on GPUs based on domain-specific CUDA functions for parallelization and speeding up computation.

The personalized nature of epilepsy and the challenges and potentials HD computing presents in this respect is explored in chapter 5. More specifically, HD models in the form of HD vectors, rather than complex structures, as in random forests or neural networks, enable simple comparison of models within and between subjects. This further leads to the possibility of combining individual models into general models, which are of high interest for future wearable monitoring, where it is not expected to be able to monitor each patient for days in order to train personalized models. Thus, due to the aforementioned practicality of generalized models, but also the strong personal character of epilepsy, we investigate and propose how HD computing can be used to study the interplay between generalized and personalized models. We propose hybrid models that consist partially of personalized and partially of generalized models. We show that this approach can improve performance and help overcome the challenges of balanced or unbalanced databases. Moreover, this approach enables knowledge transfer between different databases.

Lastly, in chapter 6, we focus on demonstrating an additional potential of HD computing, namely, addressing the 'black-box' problem of many traditional ML methods. More specifically, we show how specific types of encoding can enable feature comparison, as well as selection, as part of the HD workflow. Similarly, when the same approach is applied to channel selection, it can lead to a significant reduction in the number of required channels. This approach automatically and independently decides optimal channels for every subject. In the end, visual analytics is becoming a more important aspect of ML interpretability, and here we demonstrate how HD computing can be used to explain predictions (and confidences) of individual channels and features in time. This can be further used to detect missed seizures, noise types that cause false positives, or channels/features that might be redundant or too imprecise. It can be further used to study the seizure evolution in time, both in terms of signal characteristics and its location.

Finally, after all the analysis is done, one might ask, 'so what is the best approach that you have with hyperdimensional computing for epilepsy detection'? The answer is not so simple, but here are several recommendations that became clearer through the thesis:

- For wearable applications, there is a definite need to have as few and as simple features as possible. For this reason, features like the proposed Approximate-Zero Crossing features should be used.
- Channel selection should be made as it can lead to a significant decrease in the number of channels without a noticeable reduction in performance.

- Encoding of raw data to the HD vectors should be done in a way that maps spatial and temporal information but is also interpretable. This can further allow a better understanding of how the model works, and how to reduce the number of channels, and features or improve learning itself.
- Due to the usage of time-series cross-validation and long-term datasets, more elaborate training procedures are needed. For example, online learning with weighting novelties of new data before updating the model or allowing multiple HD centroids per class is needed.
- After models are trained, prototype class vectors can be compared, and vector bits that are the same for all classes can be removed from the vectors as they are not contributing to the prediction. This dimension reduction of HD vectors can further lower memory and computational requirements. However, this is only possible for testing and if there is no retraining of models on the device.

In the end, we hope that work done in the scope of this thesis will inspire other researchers to develop the HD computing approach further, taking into account the opportunities and challenges biosignal monitoring has. We do believe that HD computing has a lot to offer for long-term monitoring using wearable devices, especially for healthcare applications.

7.2 Future work

Encoding spatio-temporal data
First, when talking about spatial representation, there is research to be made on the optimal initialization of feature and channel base vectors. More specifically, in case of unknown exact location (such as in iEEG datasets), the only option is random initialization, but in cases with a known position on the scalp, more specific EEG-based initialization could be tested that will map 2D/3D distances between channels to the similarities between vectors. The hypothesis is that such initialization could further improve performance, but this should be examined.

Furthermore, we focused on capturing time information within the features themselves, but in the future, more effort should be put into designing inherent encoding of time information into the HD model without the need for expert-designed features that quantify time evolution.

A similar approach to embedding time information during the encoding process, as was done here with spatial information, should be analyzed in the future. In fact, some researchers utilized vector permutation [204] for encoding time information while encoding raw data. However, a more systematic analysis is necessary.

Finally, since hyperdimensional computing is highly interesting for energy-efficient hardware implementation due to parallelization possibilities, as shown in many recent works [200]–[202], one of the next steps is exploring hardware-software co-design. In particular, data encoding is the most interesting part, as traditionally encoding takes the most computational effort.

Learning with unbalanced datasets
We believe that all proposed improvements for HD learning, namely, *multi-centroid*, *iterative* and *weighted ('OnlineHD')* learning, tackle the problem in different ways and that combining two or even all three approaches could lead to even better learning capabilities. Thus, we think this is one of the future venues for the research.

Further, the influence of commonly used approaches of data undersampling and oversampling could be systematically tested on HD computing. This could potentially help design more complex undersampling and oversampling techniques while taking into account real-life implementability.

In the end, more efforts are needed to create general libraries for HD computing that will be commonly used. These libraries should be on one side focused on server and GPU applications with the goal of speeding up design-space exploration of HD algorithms, especially when used on large datasets. On the other hand, HD libraries for microcontrollers, FPGAs, etc., are also needed. For this, more collaborations with other researchers are needed.

Personalized nature
Results in chapter 5 are extremely interesting, but we would like to explore them even further. First, a wide range of datasets should be used to create personalized and generalized models. They should be compared (personalized to generalized from each database), and

generalization between different datasets could be measured (generalized models between different databases). Next, knowledge transfer should be tested using hybrid models as well. An interesting question to explore is how to combine models and knowledge from several datasets, all with different properties (data balance, number of channels, types of seizures). Whether it is possible to combine all databases to create one general model that will perform well on all of them, also remains an open question.

One more idea that was not fully explored is whether individual personalized models can be clustered into several subgroups instead of all being grouped into one general model, analogous to the previously presented *multi-centroid* approach. Indeed, there might be different types of seizures or patterns present in parts of the patient population, and an unsupervised algorithm that could perform this clustering might be interesting, both from medical and also performance improvement perspectives.

Interpretability
The presented $FeatAppend$ approach is a very simple approach that also has several limitations. Thus in the future, we hope to work on more elaborate encoding that is not restrictive on the number of channels and would also enable more than simple linear combinations of channels. This encoding would still need to keep the capability to backtrace the contributions of each feature/channel to maintain interpretability possibilities. For example, an approach from the recent paper [391] that utilizes a fully connected layer as part of the encoding process could be a potential venue.

Finally, it would be interesting to use interpretability possibilities to study the interplay between personal and generalized models. In particular, one can keep both personal and general models and compare predictions and confidences of each model in time for each of the subjects. This could be further used to understand which patterns of signals are general and which are subject-specific.

Other ideas
In the end, one more avenue of future research is using HD computing with multimodal data for epilepsy detection. The idea is to use datasets including several biosignal modalities (e.g., EEG, PPG, EDA, SKT, ACC) to study sensor fusion combined with the above-mentioned interpretability of predictions. This could not only lead to better detection performance but also a better understanding of the time component and contributions of each biosignal modality. The main challenge here is the availability of such multimodal datasets.

Lastly, but very importantly, once an optimal HD workflow with appropriate spatio-temporal encoding and elaborate (but not computationally complex) learning is designed, this should be ported to a microcontroller as part of a wearable device such as the ones proposed in the literature, e.g., e-glasses [36], or behind-the-ear [161] or in-ear EEG devices [38]. Only then the true potential of HD computing for disease and health monitoring will be properly visible.

Evolution of the experimental setup throughout the thesis

The order of presented results in the thesis does not necessarily corresponds to the order in which results have been obtained. This can lead to confusion about the consistency of experimental setups throughout the thesis. Thus in Table 7.1, an overview of data, methodologies, and main parameters used in all the research studies is visualized.

Differences exist because the need for improvements of different parts of HD workflow has been realized during the thesis. For example, first, we focused on investigating features to use (#1 in the table). Then, we wondered if encoding spatial information can bring improvement and how to do it efficiently (#2). While doing it, we realized that one of the proposed encoding approaches, 'FeatAppend', can be potentially used to perform feature selection as well (#3). However, we also realized that the learning capabilities of HD computing are much smaller if we want to use more realistic and less balanced datasets. Thus we next focused on improving the learning capabilities of HD computing by proposing a multi-centroid learning approach (#4)and then systematically comparing it to other approaches proposed in the literature (#5).

Meanwhile, while inspecting the performance of our algorithms together with neurologists, we got inspired to test features inspired by how neurologists recognized seizures, which led to the proposal of AZC features (#6). This has also led to the realization that we can remove some features and utilize AZC features instead in the future.

In this work, we also started using whole datasets (with RF models) and time-series-cross-validation. However, at that moment, we could not run the HD model on the entire CHB-MIT and SWEC-ETHZ (long) datasets and decided to work on a library utilizing GPU-s to enable this (#7). Finally, having this ability, we wondered about the influence of cross-validation as well as the amount of data used. Thus, we systematically assessed the influence of various methodological choices and data used on reported performance (#8).

In the end, we wanted to focus on the real-life usability of HD computing, e.g., utilizing generalized models as much as possible or combining them with personalized models (#9). This also inspired us to test possibilities for channel selection (#10). Finally, thinking about the interpretability of the models, we wanted to visualize predictions per channel and feature.

To make it comparable to previous feature selection work, we used the same experimental setup as in (#3).

I hope that this reasoning behind the chronological order in which studies have been done also helps in understanding the context and limitations of each study.

Chapters have been, on the other hand, grouped to point to other important aspects that are not necessarily present in various applications of HD computing, but are necessary to take into account when dealing with long-term biosignals for healthcare monitoring. More specifically, we grouped them under topics of dealing with spatio-temporal data, as well as unbalanced and personalized nature. We also focused on the need to understand the working of the models and visualize their predictions. Finally, since ultimately, models will be used by patients, we discussed the influence of our methodological choices and the performance metrics used to assess the quality of algorithms for future real-life implementations.

Table 7.1: Overview of all studies done in the scope of the thesis.

Study	Chapter	Order	Database	Data preparation	Features used
Methodological choices	Sec. 2	#8	CHB-MIT	Fact1 and Fact10, All data 1h, 4h and StoS	19 features (mAmpl, LL, 17 freq)
Features for HD computing	Sec. 3.2	#1	CHB-MIT and SWEC-ETHZ short	Fact1 to be comparable	Indiv. features and multiple features (3 and 45)
AZC features	Sec. 3.3	#6	CHB-MIT and SWEC-ETHZ long	All data 1h	6 AZC features vs 56 CLF features(mAmpl, LL, 17 freq and 37 entropy)
Spatial encoding	Sec. 3.4	#2	CHB-MIT	Fact10	47 features (mAmpl, LL, 8 rel. freq and 37 entropy)
Multi centroid	Sec. 4.2	#4	CHB-MIT	Fact1, Fact5 and Fact10	47 features (mAmpl, 8 rel. freq and 37 entropy)
Smarter learning approaches	Sec. 4.3	#5	CHB-MIT	Fact10	47 features (mAmpl, 8 rel. freq and 37 entropy)
HDtorch	Sec. 4.4	#7	CHB-MIT	Fact10 and All data 1h	19 features (mAmpl, LL, 17 freq)
			ISOLET, MNIST, UCIHAR, PAMAP	All data	Original features from databases
Personalized and generalized	Sec. 5	#9	CHB-MIT and Repomse	Fact10 and All data from Repomse	25 features (mAmpl, LL, 17 freq and 6 AZC)
Feature selection	Sec. 6.2	#3	CHB-MIT	Fact10	47 features (mAmpl, LL, 8 rel. freq and 37 entropy)
Channel selection	Sec. 6.3	#10	CHB-MIT	Fact10	47 features (mAmpl, LL, 8 rel. freq and 37 entropy)

Publications and code repositories

All the accompanying code of various studies done during this thesis is available online. The majority of the code is in the following repository:
https://github.com/esl-epfl/HyperdimensionalComputingForEpilepsy.

However, some codes are in different repositories; thus, everything is listed in detail below.

Experimental methodology for epilepsy detection
Paper: *"Importance of methodological choices in data manipulation for validating epileptic seizure detection models"*, Una Pale, Tomas Teijeiro, David Atienza

- Submitted to 2023 45th International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)
- Arxiv version: https://arxiv.org/abs/2302.10672
- Code repository: https://github.com/esl-epfl/epilepsy_performance_metrics

Encoding spatio-temporal data
Paper: *„Systematic Assessment of Hyperdimensional Computing for Epileptic Seizure Detection"*, Una Pale, Tomas Teijeiro, David Atienza

- 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), https://ieeexplore.ieee.org/document/9629648
- Arxiv version: https://arxiv.org/abs/2105.00934
- Code repository: https://github.com/esl-epfl/HyperdimensionalComputingForEpilepsy/tree/main/01_DiffFeaturesForEpilepsy

Paper: *„Approximate Zero-Crossing: A new interpretable, highly discriminative and low-complexity feature for EEG and iEEG seizure detection"*, Renato Zanetti, Una Pale, Tomas Teijeiro, David Atienza Alonso

- Journal of Neural Engineering, https://iopscience.iop.org/article/10.1088/1741-2552/aca1e4
- Code repository: https://github.com/esl-epfl/AZCfeature

Paper: *„ExG Signal Feature Selection Using Hyperdimensional Computing Encoding"*, Una Pale,

Tomas Teijeiro, David Atienza

- 2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), https://ieeexplore.ieee.org/abstract/document/9995107
- Arxiv version: https://arxiv.org/abs/2205.07654
- Code repository: https://github.com/esl-epfl/HyperdimensionalComputingForEpilepsy/tree/main/02_FeatureSelectionWithHD

Learning with unbalanced datasets
Paper: *„Multi-Centroid Hyperdimensional Computing Approach for Epileptic Seizure Detection“*, Una Pale, Tomas Teijeiro, David Atienza

- Frontiers in Neurology, 13, 1-13, 816294, https://www.frontiersin.org/articles/10.3389/fneur.2022.816294/full
- Arxiv version: https://arxiv.org/abs/2111.08463
- Code repository: https://github.com/esl-epfl/HyperdimensionalComputingForEpilepsy/tree/main/03_MultiCentroidLearning

Paper: *„Exploration of Hyperdimensional Computing Strategies for Enhanced Learning on Epileptic Seizure Detection“*, Una Pale, Tomas Teijeiro, David Atienza

- 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), https://ieeexplore.ieee.org/document/9870919
- Arxiv version: https://arxiv.org/abs/2201.09759
- Code repository: https://github.com/esl-epfl/HyperdimensionalComputingForEpilepsy/tree/main/04_LearningImprovements

Paper: *„HDTorch: Accelerating Hyperdimensional Computing with GP-GPUs for Design Space Exploration“*, William Andrew Simon, Una Pale, Tomas Teijeiro, David Atienza

- 41st IEEE/ACM International Conference on Computer-Aided Design (ICCAD), https://dl.acm.org/doi/abs/10.1145/3508352.3549475
- Arxiv version: https://arxiv.org/abs/2206.04746
- Code repository: https://pypi.org/project/hdtorch

Personalized nature
Paper: *„Combining General and Personalized Models for Epilepsy Detection with Hyperdimensional Computing“*, Una Pale, Tomas Teijeiro, David Atienza

- Submitted to 2023 Conference on Lifelong Learning Agents (CoLLAs)
- Arxiv version: https://arxiv.org/abs/2303.14745
- Code repository: https://github.com/esl-epfl/HyperdimensionalComputingForEpilepsy/tree/main/05_PersAndGenModels

Interpretability
Paper: *„ExG Signal Feature Selection Using Hyperdimensional Computing Encoding“*, Una Pale,

Tomas Teijeiro, David Atienza

- 2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), https://ieeexplore.ieee.org/abstract/document/9995107
- Arxiv version: https://arxiv.org/abs/2205.07654
- Code repository: https://github.com/esl-epfl/HyperdimensionalComputingForEpilepsy/tree/main/02_FeatureSelectionWithHD

Bibliography

[1] J. Amann *et al.*, "Explainability for artificial intelligence in healthcare: a multidisciplinary perspective", *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, p. 310, 2020. DOI: 10.1186/s12911-020-01332-6.

[2] D. Higgins and V. I. Madai, "From bit to bedside: a practical framework for artificial intelligence product development in healthcare", *Advanced Intelligent Systems*, vol. 2, no. 10, p. 2 000 052, 2020. DOI: 10.1002/aisy.202000052.

[3] K. Kourou *et al.*, "Machine learning applications in cancer prognosis and prediction", *Computational and Structural Biotechnology Journal*, vol. 13, pp. 8–17, 2015. DOI: 10.1016/j.csbj.2014.11.005.

[4] T. M. Jørgensen *et al.*, "Machine-learning classification of non-melanoma skin cancers from image features obtained by optical coherence tomography", *Skin Research and Technology Journal*, vol. 14, no. 3, pp. 364–369, 2008. DOI: 10.1111/j.1600-0846.2008.00304.x.

[5] E. Moradi *et al.*, "Machine learning framework for early MRI-based alzheimer's conversion prediction in MCI subjects", *NeuroImage*, vol. 104, pp. 398–412, 2015. DOI: 10.1016/j.neuroimage.2014.10.002.

[6] V. Gulshan *et al.*, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs", *JAMA*, vol. 316, no. 22, pp. 2402–2410, 2016. DOI: 10.1001/jama.2016.17216.

[7] G. Zaharchuk *et al.*, "Deep learning in neuroradiology", *American Journal of Neuroradiology*, 2018. DOI: 10.3174/ajnr.A5543.

[8] K. W. Johnson *et al.*, "Artificial intelligence in cardiology", *Journal of the American College of Cardiology*, vol. 71, no. 23, pp. 2668–2679, 2018. DOI: 10.1016/j.jacc.2018.03.521.

[9] P. Kukharchik *et al.*, "Vocal fold pathology detection using modified wavelet-like features and support vector machines", in *European Signal Processing Conference*, 2007, pp. 2214–2218.

[10] M. W. Libbrecht and W. S. Noble, "Machine learning applications in genetics and genomics", *Nature Reviews. Genetics*, vol. 16, no. 6, pp. 321–332, 2015. DOI: 10.1038/nrg3920.

[11] S. A. Waldman and A. Terzic, "Healthcare Evolves From Reactive to Proactive", *Clinical pharmacology and therapeutics*, vol. 105, no. 1, pp. 10–13, 2019. DOI: 10.1002/cpt.1295.

[12] R. Z. Goetzel, "Do prevention or treatment services save money? The wrong debate", *Health Affairs (Project Hope)*, vol. 28, no. 1, pp. 37–41, 2009. DOI: 10.1377/hlthaff.28.1.37.

[13] I. S. Chan and G. S. Ginsburg, "Personalized medicine: progress and promise", *Review of Genomics and Human Genetics*, vol. 12, pp. 217–244, 2011. DOI: 10.1146/annurev-genom-082410-101446.

[14] V. Puri *et al.*, "Artificial intelligence-powered decentralized framework for Internet of Things in Healthcare 4.0", *Transactions on Emerging Telecommunications Technologies*, e4245, 2021. DOI: 10.1002/ett.4245.

[15] A. Tricoli *et al.*, "Wearable and Miniaturized Sensor Technologies for Personalized and Preventive Medicine", *Advanced Functional Materials*, vol. 27, no. 15, p. 1 605 271, 2017. DOI: 10.1002/adfm.201605271.

[16] F. Mormann *et al.*, "Seizure prediction: the long and winding road", *Brain*, vol. 130, no. Pt 2, pp. 314–333, 2007. DOI: 10.1093/brain/awl241.

[17] J. F. Tellez-Zenteno *et al.*, "Psychiatric comorbidity in epilepsy: a population-based analysis", *Epilepsia*, vol. 48, no. 12, pp. 2336–2344, 2007. DOI: 10.1111/j.1528-1167.2007.01222.x.

[18] M. Ihle *et al.*, "EPILEPSIAE - a european epilepsy database", *Computer Methods and Programs in Biomedicine*, vol. 106, no. 3, pp. 127–138, 2012. DOI: 10.1016/j.cmpb.2010.08.011.

[19] D. Hirtz *et al.*, "How common are the "common" neurologic disorders?", *Neurology*, vol. 68, no. 5, pp. 326–337, 2007. DOI: 10.1212/01.wnl.0000252807.38124.a3.

[20] D. J. Thurman *et al.*, "Sudden unexpected death in epilepsy: assessing the public health burden", *Epilepsia*, vol. 55, no. 10, pp. 1479–1485, 2014. DOI: 10.1111/epi.12666.

[21] P. Kwan and M. J. Brodie, "Early identification of refractory epilepsy", *The New England Journal of Medicine*, vol. 342, no. 5, pp. 314–319, 2020. DOI: 10.1056/NEJM200002033420503.

[22] R. S. Fisher *et al.*, "Instruction manual for the ILAE 2017 operational classification of seizure types", *Epilepsia*, vol. 58, no. 4, pp. 531–542, 2017. DOI: 10.1111/epi.13671.

[23] A. Sen *et al.*, "Epilepsy in older people", *The Lancet*, vol. 395, no. 10225, pp. 735–748, 2020. DOI: 10.1016/S0140-6736(19)33064-8.

[24] K. M. Fiest *et al.*, "Prevalence and incidence of epilepsy: a systematic review and meta-analysis of international studies", *Neurology*, vol. 88, no. 3, pp. 296–303, 2017. DOI: 10.1212/WNL.0000000000003509.

[25] S. Arthurs *et al.*, "Patient and caregiver perspectives on seizure prediction", *Epilepsy & Behavior: E&B*, vol. 19, no. 3, pp. 474–477, 2010. DOI: 10.1016/j.yebeh.2010.08.010.

[26] J. Klatt *et al.*, "The EPILEPSIAE database: an extensive electroencephalography database of epilepsy patients", *Epilepsia*, vol. 53, no. 9, 2012. DOI: 10.1111/j.1528-1167.2012.03564.x.

[27] A. D. Patel *et al.*, "Patient-centered design criteria for wearable seizure detection devices", *Epilepsy & Behavior: E&B*, vol. 64, no. Pt A, 2016. DOI: 10.1016/j.yebeh.2016.09.012.

[28] A. Van de Vel *et al.*, "Automated non-EEG based seizure detection: Do users have a say?", *Epilepsy & Behavior*, vol. 62, 2016. DOI: 10.1016/j.yebeh.2016.06.029.

[29] K. Rasheed *et al.*, "Machine Learning for Predicting Epileptic Seizures Using EEG Signals: A Review", *IEEE reviews in biomedical engineering*, vol. 14, pp. 139–155, 2021. DOI: 10.1109/RBME.2020.3008792.

[30] C. Jory *et al.*, "Safe and sound? A systematic literature review of seizure detection methods for personal use", *Seizure*, vol. 36, pp. 4–15, 2016. DOI: 10.1016/j.seizure.2016.01.013.

[31] B. H. Brinkmann *et al.*, "Seizure Diaries and Forecasting With Wearables: Epilepsy Monitoring Outside the Clinic", *Frontiers in Neurology*, vol. 12, 2021.

[32] J. Verdru and W. Van Paesschen, "Wearable seizure detection devices in refractory epilepsy", *Acta Neurologica Belgica*, vol. 120, no. 6, pp. 1271–1281, 2020. DOI: 10.1007/s13760-020-01417-z.

[33] G. Giannakakis *et al.*, "Methods for seizure detection and prediction: an overview", in *Modern Electroencephalographic Assessment Techniques: Theory and Applications*, ser. Neuromethods, Springer, 2015, pp. 131–157.

[34] E. Bou Assi *et al.*, "Towards accurate prediction of epileptic seizures: a review", *Biomedical Signal Processing and Control*, vol. 34, pp. 144–157, 2017. DOI: 10.1016/j.bspc.2017.02.001.

[35] D. Callegari *et al.*, "EpiCare — a home care platform based on mobile cloud computing to assist epilepsy diagnosis", in *International Conference on Wireless Mobile Communication and Healthcare*, 2014, pp. 148–151. DOI: 10.1109/MOBIHEALTH.2014.7015931.

[36] D. Sopic *et al.*, "E-Glass: A Wearable System for Real-Time Detection of Epileptic Seizures", in *IEEE International Symposium on Circuits and Systems AISCAS*, 2018, pp. 1–5. DOI: 10.1109/ISCAS.2018.8351728.

[37] M. Guermandi *et al.*, "A wearable device for minimally-invasive behind-the-ear EEG and evoked potentials", in *IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2018, pp. 1–4. DOI: 10.1109/BIOCAS.2018.8584814.

[38] M. Guermandi *et al.*, "A wireless system for EEG acquisition and processing in an earbud form factor with 600 hours battery lifetime", in *International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2022. DOI: 10.1109/EMBC48229.2022.9871874.

[39] S. Shorvon *et al.*, Eds., *Oxford Textbook of Epilepsy and Epileptic Seizures*, Oxford Textbooks in Clinical Neurology, Oxford University Press, 2012.

[40] N. Kissani *et al.*, "Sensitivity of recordings at sphenoidal electrode site for detecting seizure onset: evidence from scalp, superficial and deep foramen ovale recordings", *Clinical Neurophysiology*, vol. 112, no. 2, pp. 232–240, 2001. DOI: 10.1016/s1388-2457(00)00531-9.

[41] H. M. de Boer *et al.*, "The global burden and stigma of epilepsy", *Epilepsy & Behavior*, vol. 12, no. 4, pp. 540–546, 2008. DOI: 10.1016/j.yebeh.2007.12.019.

[42] E. Beghi *et al.*, "Morbidity and accidents in patients with epilepsy: results of a european cohort study", *Epilepsia*, vol. 43, no. 9, pp. 1076–1083, 2002. DOI: 10.1046/j.1528-1157.2002.18701.x.

[43] J. F. Kurtzke, "Epilepsy: frequency, causes and consequences", *Archives of Neurology*, vol. 49, no. 4, p. 342, 1992. DOI: 10.1001/archneur.1992.00530280020007.

[44] R. C. Kessler *et al.*, "Accounting for comorbidity in assessing the burden of epilepsy among US adults: results from the national comorbidity survey replication (NCS-r)", *Molecular psychiatry*, vol. 17, no. 7, pp. 748–758, 2012. DOI: 10.1038/mp.2011.56.

[45] A. Gaitatzis *et al.*, "The epidemiology of the comorbidity of epilepsy in the general population", *Epilepsia*, vol. 45, no. 12, pp. 1613–1622, 2004. DOI: 10.1111/j.0013-9580.2004.17504.x.

[46] D. W. Loring *et al.*, "Determinants of quality of life in epilepsy", *Epilepsy & Behavior*, vol. 5, no. 6, pp. 976–980, 2004. DOI: 10.1016/j.yebeh.2004.08.019.

[47] J. A. Cramer *et al.*, "The influence of comorbid depression on quality of life for people with epilepsy", *Epilepsy & Behavior*, vol. 4, no. 5, 2003. DOI: 10.1016/j.yebeh.2003.07.009.

[48] S. C. Danzer, "Depression, stress, epilepsy and adult neurogenesis", *Experimental Neurology*, vol. 233, no. 1, pp. 22–32, 2012. DOI: 10.1016/j.expneurol.2011.05.023.

[49] A. M. Kanner, "Depression and epilepsy: a bidirectional relation?", *Epilepsia*, vol. 52, pp. 21–27, 2011. DOI: 10.1111/j.1528-1167.2010.02907.x.

[50] A. Thapar *et al.*, "Stress, anxiety, depression, and epilepsy: investigating the relationship between psychological factors and seizures", *Epilepsy & Behavior*, vol. 14, no. 1, pp. 134–140, 2009. DOI: 10.1016/j.yebeh.2008.09.004.

[51] M. Pompili *et al.*, "Depression, hopelessness and suicide risk among patients suffering from epilepsy", *Annali dell'Istituto Superiore Di Sanita*, vol. 43, no. 4, pp. 425–429, 2007.

[52] R. E. D. Bautista and E. Tannahill Glen, "Seizure severity is associated with quality of life independent of seizure frequency", *Epilepsy & Behavior*, vol. 16, no. 2, pp. 325–329, 2009. DOI: 10.1016/j.yebeh.2009.07.037.

[53] M. P. Kerr, "The impact of epilepsy on patients' lives", *Acta Neurologica Scandinavica. Supplementum*, no. 194, pp. 1–9, 2012. DOI: 10.1111/ane.12014.

[54] P. Perucca *et al.*, "Epilepsy treatment as a predeterminant of psychosocial ill health", *Epilepsy & Behavior: E&B*, vol. 15, 2009. DOI: 10.1016/j.yebeh.2009.03.016.

[55] G. A. Baker *et al.*, "Quality of life of people with epilepsy: a european study", *Epilepsia*, vol. 38, no. 3, pp. 353–362, 1997. DOI: 10.1111/j.1528-1157.1997.tb01128.x.

[56] J. Ross *et al.*, "European survey of the level of satisfaction of patients and physicians in the management of epilepsy in general practice", *Epilepsy & Behavior: E&B*, vol. 19, no. 1, pp. 36–42, 2010. DOI: 10.1016/j.yebeh.2010.06.002.

[57] R. H. Mattson *et al.*, "Comparison of carbamazepine, phenobarbital, phenytoin, and primidone in partial and secondarily generalized tonic-clonic seizures", *The New England Journal of Medicine*, vol. 313, no. 3, pp. 145–151, 1985. DOI: 10.1056/NEJM198507183130303.

[58] S. C. Karceski, "Seizure medications and their side effects", *Neurology*, vol. 69, no. 22, E27–E29, 2007. DOI: 10.1212/01.wnl.0000296051.34044.07.

[59] C. Luoni *et al.*, "Determinants of health-related quality of life in pharmacoresistant epilepsy: results from a large multicenter study of consecutively enrolled patients using validated quantitative assessments", *Epilepsia*, vol. 52, no. 12, pp. 2181–2191, 2011. DOI: 10.1111/j.1528-1167.2011.03325.x.

[60] L. Forsgren *et al.*, "Mortality of epilepsy in developed countries: a review", *Epilepsia*, vol. 46, pp. 18–27, s11 2005. DOI: 10.1111/j.1528-1167.2005.00403.x.

[61] T. Tomson, "Mortality in epilepsy", *Journal of Neurology*, vol. 247, no. 1, pp. 15–21, 2000. DOI: 10.1007/s004150050004.

[62] G. S. Bell and J. W. Sander, "Sudden unexpected death in epilepsy. risk factors, possible mechanisms and prevention: a reappraisal", *Acta Neurologica Taiwanica*, vol. 15, no. 2, pp. 72–83, 2006.

[63] R. Surges *et al.*, "Sudden unexpected death in epilepsy: risk factors and potential pathomechanisms", *Nature Reviews. Neurology*, vol. 5, no. 9, 2009. DOI: 10.1038/nrneurol.2009.118.

[64] W. J. Cardarelli and B. J. Smith, "The burden of epilepsy to patients and payers", *The American Journal of Managed Care*, vol. 16, no. 12, S331–336, 2010.

[65] S. Wiebe and D. C. Hesdorffer, "Epilepsy: being ill in more ways than one", *Epilepsy Currents*, vol. 7, no. 6, pp. 145–148, 2007. DOI: 10.1111/j.1535-7511.2007.00207.x.

[66] L. A. Copeland *et al.*, "Psychiatric and medical admissions observed among elderly patients with new-onset epilepsy", *BMC health services research*, vol. 11, p. 84, 2011. DOI: 10.1186/1472-6963-11-84.

[67] C. E. Begley *et al.*, "The cost of epilepsy in the united states: an estimate from population-based clinical and survey data", *Epilepsia*, vol. 41, no. 3, pp. 342–351, 2000. DOI: 10.1111/j.1528-1157.2000.tb00166.x.

[68] J. I. Ivanova *et al.*, "Economic burden of epilepsy among the privately insured in the US", *PharmacoEconomics*, vol. 28, no. 8, 2010. DOI: 10.2165/11535570-000000000.

[69] M. Pugliatti *et al.*, "Estimating the cost of epilepsy in europe: a review with economic modeling", *Epilepsia*, vol. 48, no. 12, pp. 2224–2233, 2007. DOI: 10.1111/j.1528-1167.2007.01251.x.

[70] C. Begley *et al.*, "The global cost of epilepsy: a systematic review and extrapolation", *Epilepsia*, vol. 63, no. 4, pp. 892–903, 2022. DOI: 10.1111/epi.17165.

[71] S. V. Thomas *et al.*, "Economic burden of epilepsy in india", *Epilepsia*, vol. 42, no. 8, pp. 1052–1060, 2001. DOI: 10.1046/j.1528-1157.2001.0420081052.x.

[72] P. R. Carney *et al.*, "Seizure prediction: methods", *Epilepsy & behavior*, vol. 22, S94–S101, 2011. DOI: 10.1016/j.yebeh.2011.09.001.

[73] M. R. Islam *et al.*, "Epileptic seizure focus detection from interictal electroencephalogram: a survey", *Cognitive Neurodynamics*, vol. 17, no. 1, pp. 1–23, 2023. DOI: 10.1007/s11571-022-09816-z.

[74] K. Rai *et al.*, "Features extraction for classification of focal and non-focal EEG signals", K. J. Kim, Ed., pp. 599–605, 2015. DOI: 10.1007/978-3-662-46578-3_70.

[75] E. Acar *et al.*, "Computational analysis of epileptic focus localization", IASTED International conference on Biomedical engineering, pp. 317–322, 2006.

[76] S. Fakhraei *et al.*, "Confidence in medical decision making: application in temporal lobe epilepsy data mining", in *Workshop on Data mining for medicine and healthcare (DMMH)*, 2011, pp. 60–63. DOI: 10.1145/2023582.2023593.

[77] M. Ghannad-Rezaie *et al.*, "Medical data mining using particle swarm optimization for temporal lobe epilepsy", in *IEEE International Conference on Evolutionary Computation*, 2006, pp. 761–768. DOI: 10.1109/CEC.2006.1688388.

[78] A. Mansouri *et al.*, "Online EEG seizure detection and localization", *Algorithms*, vol. 12, no. 9, p. 176, 2019. DOI: 10.3390/a12090176.

[79] G. Giannakakis *et al.*, "Focal epileptic seizures anticipation based on patterns of heart rate variability parameters", *Computer Methods and Programs in Biomedicine*, vol. 178, pp. 123–133, 2019. DOI: 10.1016/j.cmpb.2019.05.032.

[80] K. Vandecasteele *et al.*, "Automated epileptic seizure detection based on wearable ECG and PPG in a hospital environment", *Sensors*, vol. 17, no. 10, p. 2338, 2017. DOI: 10.3390/s17102338.

[81] F. Mohammadpour Touserkani *et al.*, "Photoplethysmographic evaluation of generalized tonic-clonic seizures", *Epilepsia*, vol. 61, no. 8, pp. 1606–1616, 2020. DOI: 10.1111/epi.16590.

[82] R. El Atrache *et al.*, "Photoplethysmography: a measure for the function of the autonomic nervous system in focal impaired awareness seizures", *Epilepsia*, vol. 61, no. 8, pp. 1617–1626, DOI: 10.1111/epi.16621.

[83] M. Casanovas Ortega *et al.*, "Electrodermal activity response during seizures: a systematic review and meta-analysis", *Epilepsy & Behavior*, vol. 134, p. 108 864, 2022. DOI: 10.1016/j.yebeh.2022.108864.

[84] M.-Z. Poh *et al.*, "Autonomic changes with seizures correlate with postictal EEG suppression", *Neurology*, vol. 78, no. 23, pp. 1868–1876, 2012. DOI: 10.1212/WNL.0b013e318258f7f1.

[85] G. Regalia *et al.*, "Multimodal wrist-worn devices for seizure detection and advancing research: focus on the empatica wristbands", *Epilepsy Research*, vol. 153, pp. 79–82, 2019. DOI: 10.1016/j.eplepsyres.2019.02.007.

[86] D. Cogan *et al.*, "Multi-biosignal analysis for epileptic seizure monitoring", *International Journal of Neural Systems*, vol. 27, no. 1, p. 1 650 031, 2017. DOI: 10.1142/S0129065716500313.

[87] S. Ganesan *et al.*, "EDA based automatic detection of epileptic seizures using wireless system", in *International Conference on Electronics, Communication and Computing Technologies*, 2011, pp. 47–52. DOI: 10.1109/ICECCT.2011.6077068.

[88] I. Conradsen *et al.*, "Automated algorithm for generalized tonic-clonic epileptic seizure onset detection based on sEMG zero-crossing rate", *IEEE transactions on bio-medical engineering*, vol. 59, no. 2, pp. 579–585, 2012. DOI: 10.1109/TBME.2011.2178094.

[89] J. J. Halford *et al.*, "Detection of generalized tonic–clonic seizures using surface electromyographic monitoring", *Epilepsia*, vol. 58, no. 11, pp. 1861–1869, 2017. DOI: 10.1111/epi.13897.

[90] S. Beniczky *et al.*, "Automated real-time detection of tonic-clonic seizures using a wearable EMG device", *Neurology*, vol. 90, no. 5, e428–e434, 2018. DOI: 10.1212/WNL.0000000000004893.

[91] M. Valderrama *et al.*, "Patient-specific seizure prediction using a multi-feature and multi-modal EEG-ECG classification", in *Mediterranean Conference on Medical and Biological Engineering and Computing*, 2010, pp. 77–80. DOI: 10.1007/978-3-642-13039-7_20.

[92] I. Mporas *et al.*, "Seizure detection using EEG and ECG signals for computer-based monitoring, analysis and management of epileptic patients", *Expert Systems with Applications*, vol. 42, no. 6, pp. 3227–3233, 2015. DOI: 10.1016/j.eswa.2014.12.009.

[93] W. A. Chaovalitwongse *et al.*, "Electroencephalogram (EEG) time series classification: applications in epilepsy", *Annals of Operations Research*, vol. 148, no. 1, pp. 227–250, 2006. DOI: 10.1007/s10479-006-0076-x.

[94] C. Guerrero-Mosquera *et al.*, "New feature extraction approach for epileptic EEG signal detection using time-frequency distributions", *Medical & Biological Engineering & Computing*, vol. 48, no. 4, pp. 321–330, 2010. DOI: 10.1007/s11517-010-0590-5.

[95] N. Koolen *et al.*, "Line length as a robust method to detect high-activity events: automated burst detection in premature EEG recordings", *Clinical Neurophysiology*, vol. 125, no. 10, pp. 1985–1994, 2014. DOI: 10.1016/j.clinph.2014.02.015.

[96] R. Esteller *et al.*, "Line length: an efficient feature for seizure onset detection", in *International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2001. DOI: 10.1109/IEMBS.2001.1020545.

[97] L. Logesparan *et al.*, "The impact of signal normalization on seizure detection using line length features", *Medical & Biological Engineering & Computing*, vol. 53, no. 10, pp. 929–942, 2015. DOI: 10.1007/s11517-015-1303-x.

[98] K. Gadhoumi *et al.*, "Discriminating preictal and interictal states in patients with temporal lobe epilepsy using wavelet analysis of intracerebral EEG", *Clinical Neurophysiology*, vol. 123, no. 10, pp. 1906–1916, 2012. DOI: 10.1016/j.clinph.2012.03.001.

[99] U. Pale *et al.*, "Exploration of hyperdimensional computing strategies for enhanced learning on epileptic seizure detection", in *International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2022, pp. 4076–4082. DOI: 10.1109/EMBC48229.2022.9870919.

[100] A. Sharmila and P. Geethanjali, "DWT based detection of epileptic seizure from EEG signals using naive bayes and k-NN classifiers", *IEEE Access*, vol. 4, pp. 7716–7727, 2016. DOI: 10.1109/ACCESS.2016.2585661.

[101] H. U. Amin *et al.*, "Feature extraction and classification for EEG signals using wavelet transform and machine learning techniques", *Australasian Physical & Engineering Sciences in Medicine*, vol. 38, no. 1, pp. 139–149, 2015. DOI: 10.1007/s13246-015-0333-x.

[102] E. Alickovic *et al.*, "Performance evaluation of empirical mode decomposition, discrete wavelet transform, and wavelet packed decomposition for automated epileptic seizure detection and prediction", *Biomedical Signal Processing and Control*, vol. 39, pp. 94–102, 2018. DOI: 10.1016/j.bspc.2017.07.022.

[103] M. Mursalin *et al.*, "Automated epileptic seizure detection using improved correlation-based feature selection with random forest classifier", *Neurocomputing*, vol. 241, pp. 204–214, 2017. DOI: 10.1016/j.neucom.2017.02.053.

[104] A. Shahid *et al.*, "Epileptic seizure detection using the singular values of EEG signals", in *ICME International Conference on Complex Medical Engineering*, 2013, pp. 652–655. DOI: 10.1109/ICCME.2013.6548330.

[105] A. K. Tafreshi *et al.*, "Epileptic seizure detection using empirical mode decomposition", in *IEEE International Symposium on Signal Processing and Information Technology*, 2008, pp. 238–242. DOI: 10.1109/ISSPIT.2008.4775717.

[106] L. Orosco *et al.*, "An epileptic seizures detection algorithm based on the empirical mode decomposition of EEG", *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, vol. 2009, pp. 2651–2654, 2009. DOI: 10.1109/IEMBS.2009.5332861.

[107] O. Stojanović *et al.*, "Predicting epileptic seizures using nonnegative matrix factorization", *PloS One*, vol. 15, no. 2, e0228025, 2020. DOI: 10.1371/journal.pone.0228025.

[108] S. Xie and S. Krishnan, "Signal decomposition by multi-scale PCA and its applications to long-term EEG signal classification", in *IEEE/ICME International Conference on Complex Medical Engineering*, 2011, pp. 532–537. DOI: 10.1109/ICCME.2011.5876798.

[109] S. M. R. Miri and A. M. Nasrabadi, "A new seizure prediction method based on return map", in *Iranian Conference of Biomedical Engineering (ICBME)*, 2011, pp. 244–248. DOI: 10.1109/ICBME.2011.6168565.

[110] J. R. Williamson *et al.*, "Seizure prediction using EEG spatiotemporal correlation structure", *Epilepsy & Behavior*, vol. 25, no. 2, pp. 230–238, 2012. DOI: 10.1016/j.yebeh.2012.07.007.

[111] F. Forooghifar *et al.*, *Self-Aware Anomaly-Detection for Epilepsy Monitoring on Low-Power Wearable Electrocardiographic Devices*. 2021, 1 p. DOI: 10.1109/AICAS51828.2021.9458555.

[112] J. Gotman, "Automatic recognition of epileptic seizures in the EEG", *Electroencephalography and Clinical Neurophysiology*, vol. 54, no. 5, 1982. DOI: 10.1016/0013-4694(82)90038-4.

[113] H. Qu and J. Gotman, "Improvement in seizure detection performance by automatic adaptation to the EEG of each patient", *Electroencephalography and Clinical Neurophysiology*, vol. 86, no. 2, pp. 79–87, 1993. DOI: 10.1016/0013-4694(93)90079-b.

[114] U. Orhan *et al.*, "EEG signals classification using the k-means clustering and a multilayer perceptron neural network model", *Expert Systems with Applications*, vol. 38, no. 10, pp. 13 475–13 481, 2011. DOI: 10.1016/j.eswa.2011.04.149.

[115] M. Li *et al.*, "Automatic epilepsy detection using wavelet-based nonlinear analysis and optimized SVM", *Biocybernetics and Biomedical Engineering*, vol. 36, no. 4, pp. 708–718, 2016. DOI: 10.1016/j.bbe.2016.07.004.

[116] D. S. K. Satapathy, "Weighted majority voting based ensemble of classifiers using different machine learning techniques for classification of EEG signal to detect epileptic seizure", *Informatica (Slovenia)*, vol. 41, no. 1, pp. 99–110, 2017.

[117] A. Shoeb and J. Guttag, "Application of machine learning to epileptic seizure detection", in *International Conference on International Conference on Machine Learning*, ser. ICML'10, Omnipress, 2010, pp. 975–982.

[118] A. Dorai and K. Ponnambalam, "Automated epileptic seizure onset detection", in *International Conference on Autonomous and Intelligent Systems, AIS*, 2010, pp. 1–4. DOI: 10.1109/AIS.2010.5547053.

[119] J. Birjandtalab *et al.*, "Unsupervised EEG analysis for automated epileptic seizure detection", in *First International Workshop on Pattern Recognition*, vol. 10011, SPIE, 2016, pp. 124–128. DOI: 10.1117/12.2243622.

[120] S. Raghu and N. Sriraam, "Classification of focal and non-focal EEG signals using neighborhood component analysis and machine learning algorithms", *Expert Systems with Applications*, vol. 113, pp. 18–32, 2018. DOI: 10.1016/j.eswa.2018.06.031.

[121] K. Polat and S. Güneş, "Classification of epileptiform EEG using a hybrid system based on decision tree classifier and fast fourier transform", *Applied Mathematics and Computation*, vol. 187, no. 2, pp. 1017–1026, DOI: 10.1016/j.amc.2006.09.022.

[122] C.-L. Chen *et al.*, "Application of chaos theory and data mining to seizure detection of epilepsy", in *International Conference on Machine Learning and Computing IACSIT*, 2012.

[123] J. Li and H. Liu, "Ensembles of cascading trees", in *IEEE International Conference on Data Mining*, 2003, pp. 585–588. DOI: 10.1109/ICDM.2003.1250983.

[124] T. K. Ho, "The random subspace method for constructing decision forests", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, 1998. DOI: 10.1109/34.709601.

[125] M. Zabihi *et al.*, "Patient-specific epileptic seizure detection in long-term EEG recording in paediatric patients with intractable seizures", in *IET Intelligent Signal Processing Conference (ISP)*, 2013, pp. 1–7. DOI: 10.1049/cp.2013.2060.

[126] M.-P. Hosseini *et al.*, "Random ensemble learning for EEG classification", *Artificial Intelligence in Medicine*, vol. 84, pp. 146–158, 2018. DOI: 10.1016/j.artmed.2017.12.004.

[127] A. T. Tzallas *et al.*, "Automatic seizure detection based on time-frequency analysis and artificial neural networks", *Computational Intelligence and Neuroscience*, vol. 2007, p. 80 510, DOI: 10.1155/2007/80510.

[128] P. Fergus *et al.*, "Automatic epileptic seizure detection using scalp EEG and advanced artificial intelligence techniques", *BioMed Research International*, e986736, 2015. DOI: 10.1155/2015/986736.

[129] A. Subasi *et al.*, "Epileptic seizure detection using hybrid machine learning methods", *Neural Computing and Applications*, vol. 31, no. 1, pp. 317–325, 2019. DOI: 10.1007/s00521-017-3003-y.

[130] M. Islam and H. Giggins, "Knowledge discovery through SysFor: a systematically developed forest of multiple decision trees", *Conferences in Research and Practice in Information Technology Series*, vol. 121, 2012.

[131] U. R. Acharya *et al.*, "Automated diagnosis of epileptic EEG using entropies", *Biomedical Signal Processing and Control*, vol. 7, no. 4, pp. 401–408, 2012. DOI: 10.1016/j.bspc.2011.07.007.

[132] H. Khan *et al.*, "Focal onset seizure prediction using convolutional networks", *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 9, 2018. DOI: 10.1109/TBME.2017.2785401.

[133] N. D. Truong *et al.*, "Convolutional neural networks for seizure prediction using intracranial and scalp electroencephalogram", *Neural Networks*, vol. 105, pp. 104–111, 2018. DOI: 10.1016/j.neunet.2018.04.018.

[134] R. Hussein *et al.*, "Human intracranial EEG quantitative analysis and automatic feature learning for epileptic seizure prediction", *ArXiv*, 2019.

[135] S. Muhammad Usman *et al.*, "Epileptic seizures prediction using deep learning techniques", *IEEE Access*, vol. 8, pp. 39 998–40 007, 2020. DOI: 10.1109/ACCESS.2020.2976866.

[136] K. M. Tsiouris *et al.*, "A long short-term memory deep learning network for the prediction of epileptic seizures using EEG signals", *Computers in Biology and Medicine*, vol. 99, pp. 24–37, 2018. DOI: 10.1016/j.compbiomed.2018.05.019.

[137] X. Yao *et al.*, "A robust deep learning approach for automatic classification of seizures against non-seizures", *Biomedical Signal Processing and Control*, vol. 64, p. 102 215, DOI: 10.1016/j.bspc.2020.102215.

[138] L. Vidyaratne *et al.*, "Deep recurrent neural network for seizure detection", *International Joint Conference on Neural Networks (IJCNN)*, 2016. DOI: 10.1109/IJCNN.2016.7727334.

[139] A. Antoniades *et al.*, "Deep learning for epileptic intracranial EEG data", in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016. DOI: 10.1109/MLSP.2016.7738824.

[140] M. K. Siddiqui *et al.*, "A review of epileptic seizure detection using machine learning classifiers", *Brain Informatics*, vol. 7, no. 1, p. 5, 2020. DOI: 10.1186/s40708-020-00105-1.

[141] I. Ahmad *et al.*, "EEG-based epileptic seizure detection via machine/deep learning approaches: a systematic review", *Computational Intelligence and Neuroscience*, vol. 2022, p. 6 486 570, 2022. DOI: 10.1155/2022/6486570.

[142] A. Shoeibi *et al.*, "Epileptic seizures detection using deep learning techniques: a review", *International Journal of Environmental Research and Public Health*, vol. 18, no. 11, p. 5780, 2021. DOI: 10.3390/ijerph18115780.

[143] M. N. Adnan and M. Z. Islam, "ForEx++: a new framework for knowledge discovery from decision forests", *Australasian Journal of Information Systems*, vol. 21, 2017. DOI: 10.3127/ajis.v21i0.1539.

[144] X. Wang *et al.*, "Detection analysis of epileptic EEG using a novel random forest model combined with grid search optimization", *Frontiers in Human Neuroscience*, vol. 13, 2019.

[145] K. D. Tzimourta *et al.*, "A robust methodology for classification of epileptic seizures in EEG signals", *Health and Technology*, vol. 9, no. 2, 2019. DOI: 10.1007/s12553-018-0265-z.

[146] M. A. Pinto-Orellana and F. R. Cerqueira, "Patient-specific epilepsy seizure detection using random forest classification over one-dimension transformed EEG data", in *Intelligent Systems Design and Applications*, 2017, pp. 519–528. DOI: 10.1007/978-3-319-53480-0_51.

[147] N. D. Truong *et al.*, "Supervised learning in automatic channel selection for epileptic seizure detection", *Expert Systems with Applications*, vol. 86, pp. 199–207, 2017. DOI: 10.1016/j.eswa.2017.05.055.

[148] N. D. Truong *et al.*, "Epileptic seizure forecasting with generative adversarial networks", *IEEE Access*, vol. 7, pp. 143 999–144 009, 2019. DOI: 10.1109/ACCESS.2019.2944691.

[149] A. M. Abdelhameed and M. Bayoumi, "Semi-supervised deep learning system for epileptic seizures onset prediction", in *IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2018, pp. 1186–1191. DOI: 10.1109/ICMLA.2018.00191.

[150] A. Amirshahi *et al.*, "M2D2: Maximum-Mean-Discrepancy Decoder for Temporal Localization of Epileptic Brain Activities", *IEEE Journal of Biomedical and Health Informatics*, vol. PP, 2022. DOI: 10.1109/JBHI.2022.3208780.

[151] A. Schulze-Bonhage, "Seizure prediction: Time for new, multimodal and ultra-long-term approaches", *Clinical Neurophysiology*, vol. 133, 2022. DOI: 10.1016/j.clinph.2021.10.006.

[152] D. Cogan *et al.*, "Personalization of NonEEG-based seizure detection systems", in *2016 International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2016, pp. 6349–6352. DOI: 10.1109/EMBC.2016.7592180.

[153] T. De Cooman *et al.*, "Personalizing Heart Rate-Based Seizure Detection Using Supervised SVM Transfer Learning", *Frontiers in Neurology*, vol. 11, 2020.

[154] M. Poh *et al.*, "Continuous monitoring of electrodermal activity during epileptic seizures using a wearable sensor", in *International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2010, pp. 4415–4418. DOI: 10.1109/IEMBS.2010.5625988.

[155] S. Beniczky *et al.*, "Detection of generalized tonic–clonic seizures by a wireless wrist accelerometer: A prospective, multicenter study", *Epilepsia*, vol. 54, no. 4, e58–e61, 2013. DOI: https://doi.org/10.1111/epi.12120.

[156] S. K. Simblett *et al.*, "Patient perspectives on the acceptability of mHealth technology for remote measurement and management of epilepsy: a qualitative analysis", *Epilepsy & Behavior: E&B*, vol. 97, pp. 123–129, 2019. DOI: 10.1016/j.yebeh.2019.05.035.

[157] A. Emami *et al.*, "Seizure detection by convolutional neural network-based analysis of scalp electroencephalography plot images", *Neuroimage Clin.*, vol. 22, p. 101 684, 2019. DOI: 10.1016/j.nicl.2019.101684.

[158] S. Ghosh-Dastidar *et al.*, "Principal Component Analysis-Enhanced Cosine Radial Basis Function Neural Network for Robust Epilepsy and Seizure Detection", *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 2, pp. 512–518, 2008. DOI: 10.1109/TBME.2007.905490.

[159] E. Bruno *et al.*, "Wearable devices for seizure detection: practical experiences and recommendations from the wearables for epilepsy and research (WEAR) international study group", *Epilepsia*, vol. 62, no. 10, pp. 2307–2321, DOI: 10.1111/epi.17044.

[160] E. D. McKenzie *et al.*, "Validation of a smartphone-based EEG among people with epilepsy: a prospective study", *Scientific Reports*, vol. 7, p. 45 567, 2017. DOI: 10.1038/srep45567.

[161] L. Swinnen *et al.*, "Accurate detection of typical absence seizures in adults and children using a two-channel electroencephalographic wearable behind the ears", *Epilepsia*, vol. 62, no. 11, pp. 2741–2752, 2021. DOI: 10.1111/epi.17061.

[162] S.-K. Lin *et al.*, "An ultra-low power smart headband for real-time epileptic seizure detection", *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 6, pp. 1–10, 2018. DOI: 10.1109/JTEHM.2018.2861882.

[163] K. Vandecasteele *et al.*, "Visual seizure annotation and automated seizure detection using behind-the-ear electroencephalographic channels", *Epilepsia*, vol. 61, no. 4, pp. 766–775, 2020. DOI: 10.1111/epi.16470.

[164] Y. Gu *et al.*, "Comparison between scalp EEG and behind-the-ear EEG for development of a wearable seizure detection system for patients with focal epilepsy", *Sensors*, vol. 18, no. 1, 2018. DOI: 10.3390/s18010029.

[165] I. C. Zibrandtsen *et al.*, "Ear-EEG detects ictal and interictal abnormalities in focal and generalized epilepsy - a comparison with scalp EEG monitoring", *Clinical Neurophysiology*, vol. 128, no. 12, pp. 2454–2461, 2017. DOI: 10.1016/j.clinph.2017.09.115.

[166] N. Pham *et al.*, "WAKE: a behind-the-ear wearable system for microsleep detection", in *International Conference on Mobile Systems, Applications, and Services*, 2020, pp. 404–418. DOI: 10.1145/3386901.3389032.

[167] S. D. Jørgensen *et al.*, "Ear-EEG-based sleep scoring in epilepsy: a comparison with scalp-EEG", *Journal of Sleep Research*, vol. 29, no. 6, e12921, 2020. DOI: 10.1111/jsr.12921.

[168] C. Á. Szabó *et al.*, "Electromyography-based seizure detector: preliminary results comparing a generalized tonic-clonic seizure detection algorithm to video-EEG recordings", *Epilepsia*, vol. 56, no. 9, pp. 1432–1437, 2015. DOI: 10.1111/epi.13083.

[169] P. Meritam *et al.*, "User-based evaluation of applicability and usability of a wearable accelerometer device for detecting bilateral tonic–clonic seizures: a field study", *Epilepsia*, vol. 59, pp. 48–52, S1 2018. DOI: 10.1111/epi.14051.

[170] A. L. Patterson *et al.*, "SmartWatch by SmartMonitor: assessment of seizure detection efficacy for various seizure types in children, a large prospective single-center study", *Pediatric Neurology*, vol. 53, no. 4, pp. 309–311, 2015. DOI: 10.1016/j.pediatrneurol.2015.07.002.

[171] E. Schulc *et al.*, "Measurement and quantification of generalized tonic–clonic seizures in epilepsy patients by means of accelerometry—an explorative study", *Epilepsy Research*, vol. 95, no. 1, pp. 173–183, 2011. DOI: 10.1016/j.eplepsyres.2011.02.010.

[172] S. Kusmakar *et al.*, "Automated detection of convulsive seizures using a wearable accelerometer device", *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 2, pp. 421–432, 2019. DOI: 10.1109/TBME.2018.2845865.

[173] K. A. Myers *et al.*, "Heart rate variability measurement in epilepsy: how can we move from research to clinical practice?", *Epilepsia*, vol. 59, no. 12, 2018. DOI: 10.1111/epi.14587.

[174] S. Sivathamboo *et al.*, "Cardiorespiratory and autonomic function in epileptic seizures: a video-EEG monitoring study", *Epilepsy & Behavior: E&B*, vol. 111, p. 107 271, 2020. DOI: 10.1016/j.yebeh.2020.107271.

[175] U. Kramer *et al.*, "A novel portable seizure detection alarm system: preliminary results", *Journal of Clinical Neurophysiology*, vol. 28, no. 1, p. 36, 2011. DOI: 10.1097/WNP.0b013e3182051320.

[176] J. Arends *et al.*, "Multimodal nocturnal seizure detection in a residential care setting: a long-term prospective trial", *Neurology*, vol. 91, no. 21, e2010–e2019, 2018. DOI: 10.1212/WNL.0000000000006545.

[177] M.-Z. Poh *et al.*, "Convulsive seizure detection using a wrist-worn electrodermal activity and accelerometry biosensor", *Epilepsia*, vol. 53, no. 5, e93–97, 2012. DOI: 10.1111/j.1528-1167.2012.03444.x.

[178] E. Bruno *et al.*, "Day and night comfort and stability on the body of four wearable devices for seizure detection: a direct user-experience", *Epilepsy & Behavior: E&B*, vol. 112, p. 107 478, 2020. DOI: 10.1016/j.yebeh.2020.107478.

[179] I. Sim, "Mobile devices and health", *New England Journal of Medicine*, vol. 381, no. 10, pp. 956–968, 2019. DOI: 10.1056/NEJMra1806949.

[180] P. Kanerva, "Hyperdimensional Computing: An Introduction to Computing in Distributed Representation with High-Dimensional Random Vectors", *Cogn. Comput.*, vol. 1, no. 2, pp. 139–159, 2009. DOI: 10.1007/s12559-009-9009-8.

[181] P. Smolensky, "Tensor product variable binding and the representation of symbolic structures in connectionist systems", *Artificial Intelligence*, vol. 46, no. 1, pp. 159–216, 1990. DOI: 10.1016/0004-3702(90)90007-M.

[182] K. Schlegel *et al.*, "A comparison of vector symbolic architectures", *Artificial Intelligence Review*, vol. 55, no. 6, pp. 4523–4555, 2022. DOI: 10.1007/s10462-021-10110-3.

[183] B. Emruli *et al.*, "Analogical mapping and inference with binary spatter codes and sparse distributed memory", in *International Joint Conference on Neural Networks (IJCNN)*, 2013, pp. 1–8. DOI: 10.1109/IJCNN.2013.6706829.

[184] D. A. Rachkovskij and E. M. Kussul, "Binding and normalization of binary sparse distributed representations by context-dependent thinning", *Neural Computation*, vol. 13, no. 2, pp. 411–452, 2001. DOI: 10.1162/089976601300014592.

[185] L. Ge and K. Parhi, "Classification using hyperdimensional computing: a review", *arXiv*, 2020. DOI: 10.1109/MCAS.2020.2988388.

[186] J. Kang *et al.*, "Xcelhd: an efficient gpu-powered hyperdimensional computing with parallelized training", *Asia and South Pacific Design Automation Conference (ASPDAC)*, 2022.

[187] M. Imani *et al.*, "VoiceHD: Hyperdimensional Computing for Efficient Speech Recognition", in *IEEE International Conference on Rebooting Computing (ICRC)*, IEEE, 2017, pp. 1–8. DOI: 10.1109/ICRC.2017.8123650.

[188] M. Imani *et al.*, "Hierarchical hyperdimensional computing for energy efficient classification", *Design Automation Conference (DAC)*, 2018.

[189] R. W. Gayler, "Vector symbolic architectures answer jackendoff's challenges for cognitive neuroscience", in *arXiv*, 2022. DOI: 10.48550/arXiv.cs/0412059.

[190] A. Rahimi *et al.*, "Efficient biosignal processing using hyperdimensional computing: network templates for combined learning and classification of ExG signals", *Proceedings of the IEEE*, vol. 107, no. 1, pp. 123–143, 2019. DOI: 10.1109/JPROC.2018.2871163.

[191] M. Imani *et al.*, "SemiHD: Semi-Supervised Learning Using Hyperdimensional Computing", in *IEEE International Conference on Computer-Aided Design (ICCAD)*, 2019, pp. 1–8. DOI: 10.1109/ICCAD45719.2019.8942165.

[192] A. Hernandez-Cane *et al.*, "Onlinehd: robust, efficient, and single-pass online learning using hyperdimensional system", *Design Automation and Test in Europe Conference (DATE)*, 2021.

[193] A. Moin *et al.*, "A wearable biosensing system with in-sensor adaptive machine learning for hand gesture recognition", *Nature Electronics*, vol. 4, no. 1, pp. 54–63, 2021. DOI: 10.1038/s41928-020-00510-8.

[194] S. Benatti *et al.*, "Online Learning and Classification of EMG-Based Gestures on a Parallel Ultra-Low Power Platform Using Hyperdimensional Computing", *IEEE Transactions on Biomedical Circuits and Systems*, vol. 13, no. 3, pp. 516–528, 2019. DOI: 10.1109/TBCAS.2019.2914476.

[195] M. Imani *et al.*, "A Framework for Collaborative Learning in Secure High-Dimensional Space", in *CLOUD Conference*, 2019, pp. 435–446. DOI: 10.1109/CLOUD.2019.00076.

[196] U. Pale *et al.*, "Multi-Centroid Hyperdimensional Computing Approach for Epileptic Seizure Detection", *Frontiers in Neurology*, vol. 13, 2022. DOI: 10.3389/fneur.2022.816294.

[197] A. Rahimi *et al.*, "A robust and energy-efficient classifier using brain-inspired hyperdimensional computing", *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2016.

[198] Y. Kim *et al.*, "GenieHD: efficient DNA pattern matching accelerator using hyperdimensional computing", in *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2020, pp. 115–120. DOI: 10.23919/DATE48585.2020.9116397.

[199] S. Salamat *et al.*, "Accelerating hyperdimensional computing on FPGAs by exploiting computational reuse", *IEEE Transactions on Computers*, vol. 69, no. 8, pp. 1159–1171, 2020. DOI: 10.1109/TC.2020.2992662.

[200] S. Salamat *et al.*, "F5-HD: Fast Flexible FPGA-based Framework for Refreshing Hyperdimensional Computing", in *ACM SIGDA Conference*, 2019, pp. 53–62. DOI: 10.1145/3289602.3293913.

[201] M. Imani *et al.*, "Revisiting HyperDimensional Learning for FPGA and Low-Power Architectures", in *IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, 2021. DOI: 10.1109/HPCA51647.2021.00028.

[202] G. Karunaratne *et al.*, "In-memory hyperdimensional computing", *arXiv:1906.01548*, 2019.

[203] H. Li *et al.*, "Hyperdimensional computing with 3d VRRAM in-memory kernels: device-architecture co-design for energy-efficient, error-resilient language recognition", in *2016 IEEE International Electron Devices Meeting (IEDM)*, 2016. DOI: 10.1109/IEDM.2016.7838428.

[204] F. Asgarinejad *et al.*, "Detection of Epileptic Seizures from Surface EEG Using Hyperdimensional Computing", in *International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2020, pp. 536–540. DOI: 10.1109/EMBC44109.2020.9175328.

[205] M. Imani *et al.*, "SparseHD: Algorithm-Hardware Co-optimization for Efficient High-Dimensional Computing", in *IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM)*, 2019, pp. 190–198. DOI: 10.1109/FCCM.2019.00034.

[206] A. Burrello *et al.*, "One-shot Learning for iEEG Seizure Detection Using End-to-end Binary Operations: Local Binary Patterns with Hyperdimensional Computing", *IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2018. DOI: 10.1109/BIOCAS.2018.8584751.

[207] A. Joshi *et al.*, *Language Geometry Using Random Indexing*. 2017, vol. 10106, 265 pp.

[208] A. Rahimi *et al.*, "Hyperdimensional computing nanosystem", *ArXiv*, 2018.

[209] T. Bandaragoda *et al.*, "Trajectory clustering of road traffic in urban environments using incremental machine learning in combination with hyperdimensional computing", in *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 1664–1670. DOI: 10.1109/ITSC.2019.8917320.

[210] M. Imani *et al.*, "HDNA: energy-efficient DNA sequencing using hyperdimensional computing", in *IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, 2018, pp. 271–274. DOI: 10.1109/BHI.2018.8333421.

[211] P. Alonso *et al.*, "HyperEmbed: tradeoffs between resources and performance in NLP tasks with hyperdimensional computing enabled embedding of n-gram statistics", in *International Joint Conference on Neural Networks (IJCNN)*, 2021, pp. 1–9. DOI: 10.1109/IJCNN52387.2021.9534359.

[212] D. Kleyko *et al.*, "Hyperdimensional computing in industrial systems: the use-case of distributed fault isolation in a power plant", *IEEE Access*, vol. 6, pp. 30 766–30 777, 2018. DOI: 10.1109/ACCESS.2018.2840128.

[213] H.-S. Kim, "HDM: hyper-dimensional modulation for robust low-power communications", in *IEEE International Conference on Communications (ICC)*, 2018, pp. 1–6. DOI: 10.1109/ICC.2018.8422472.

[214] D. Kleyko *et al.*, "Holographic graph neuron: a bioinspired architecture for pattern processing", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 6, pp. 1250–1262, 2017. DOI: 10.1109/TNNLS.2016.2535338.

[215] A. X. Manabat *et al.*, "Performance analysis of hyperdimensional computing for character recognition", in *International Symposium on Multimedia and Communication Technology (ISMAC)*, 2019, pp. 1–5. DOI: 10.1109/ISMAC.2019.8836136.

[216] E. M. Kussul *et al.*, "Permutation coding technique for image recognition systems", *IEEE Transactions on Neural Networks*, vol. 17, no. 6, pp. 1566–1579, 2006. DOI: 10.1109/TNN.2006.880676.

[217] A. Curtidor *et al.*, "Analysis of random local descriptors in face recognition", *Electronics*, vol. 10, no. 11, p. 1358, 2021. DOI: 10.3390/electronics10111358.

[218] A. Mitrokhin *et al.*, "Learning sensorimotor control with neuromorphic sensors: toward hyperdimensional active perception", *Science Robotics*, vol. 4, no. 30, 2019. DOI: 10.1126/scirobotics.aaw6736.

[219] R. W. Gayler and S. Levy, "A distributed basis for analogical mapping", 2009.

[220] M. Nickel *et al.*, "Holographic embeddings of knowledge graphs", in *AAAI Conference on Artificial Intelligence*, 2016, pp. 1955–1961.

[221] Y. Ma *et al.*, "Holistic representations for memorization and inference", in *Conference on Uncertainty in Artificial Intelligence*, 2018.

[222] P. Poduval *et al.*, "GrapHD: graph-based hyperdimensional memorization for brain-like cognitive learning", *Frontiers in Neuroscience*, vol. 16, 2022.

[223] E. P. Frady *et al.*, "Resonator networks, 1: an efficient solution for factoring high-dimensional, distributed representations of data structures", *Neural Computation*, vol. 32, no. 12, pp. 2311–2331, 2020. DOI: 10.1162/neco_a_01331.

[224] M. Imani *et al.*, "QuantHD: a quantization framework for hyperdimensional computing", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2020. DOI: 10.1109/TCAD.2019.2954472.

[225] J. Morris *et al.*, "CompHD: efficient hyperdimensional computing using model compression", *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2019. DOI: 10.1109/ISLPED.2019.8824908.

[226] Z. Yan *et al.*, *Efficient hyperdimensional computing*, 2023. DOI: 10.48550/arXiv.2301.10902.

[227] M. Imani *et al.*, "AdaptHD: Adaptive Efficient Training for Brain-Inspired Hyperdimensional Computing", *IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2019. DOI: 10.1109/BIOCAS.2019.8918974.

[228] E. Osipov *et al.*, *Hyperseed: unsupervised learning with vector symbolic architectures*, 2022. DOI: 10.48550/arXiv.2110.08343.

[229] C.-Y. Hsieh *et al.*, "FL-HDC: hyperdimensional computing design for the application of federated learning", in *IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, 2021, pp. 1–5. DOI: 10.1109/AICAS51828.2021.9458526.

[230] G. Karunaratne *et al.*, "Robust high-dimensional memory-augmented neural networks", *Nature Communications*, 2021. DOI: 10.1038/s41467-021-22364-0.

[231] A. Moin *et al.*, "An EMG gesture recognition system with flexible high-density sensors and brain-inspired high-dimensional classifier", in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2018, pp. 1–5. DOI: 10.1109/ISCAS.2018.8351613.

[232] A. Moin *et al.*, "Analysis of contraction effort level in EMG-based gesture recognition using hyperdimensional computing", in *IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2019, pp. 1–4. DOI: 10.1109/BIOCAS.2019.8919214.

[233] A. Rahimi *et al.*, "Hyperdimensional Computing for Blind and One-Shot Classification of EEG Error-Related Potentials", *Mobile Networks and Applications*, vol. 25, no. 5, pp. 1958–1969, 2020. DOI: 10.1007/s11036-017-0942-6.

[234] M. Hersche *et al.*, "Exploring embedding methods in binary hyperdimensional computing: a case study for motor-imagery based brain-computer interfaces", *ArXiv*, 2018.

[235] D. Kleyko *et al.*, "A hyperdimensional computing framework for analysis of cardiorespiratory synchronization during paced deep breathing", *IEEE Access*, vol. PP, pp. 1–1, 2019. DOI: 10.1109/ACCESS.2019.2904311.

[236] E.-J. Chang *et al.*, "Hyperdimensional Computing-based Multimodality Emotion Recognition with Physiological Signals", *IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, 2019. DOI: 10.1109/AICAS.2019.8771622.

[237] A. Menon *et al.*, "Efficient emotion recognition using hyperdimensional computing with combinatorial channel encoding and cellular automata", *arXiv:2104.02804*, 2021.

[238] N. Watkinson *et al.*, *Class-modeling of septic shock with hyperdimensional computing*, 2021. DOI: 10.1101/2021.05.21.21257481.

[239] A. Burrello *et al.*, "Hyperdimensional Computing With Local Binary Patterns: One-Shot Learning of Seizure Onset and Identification of Ictogenic Brain Regions Using Short-Time iEEG Recordings", *T-BME*, 2020. DOI: 10.1109/TBME.2019.2919137.

[240] A. Burrello *et al.*, "An Ensemble of Hyperdimensional Classifiers: Hardware-Friendly Short-Latency Seizure Detection With Automatic iEEG Electrode Selection", *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 4, pp. 935–946, 2021. DOI: 10.1109/JBHI.2020.3022211.

[241] R. Zanetti *et al.*, "Robust Epileptic Seizure Detection on Wearable Systems with Reduced False-Alarm Rate", *2020 International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2020. DOI: 10.1109/EMBC44109.2020.9175339.

[242] A. H. Shoeb, "Application of machine learning to epileptic seizure onset detection and treatment", Thesis, MIT, 2009.

[243] Goldberger Ary L. *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet", *Circulation*, vol. 101, no. 23, e215–e220, 2000. DOI: 10.1161/01.CIR.101.23.e215.

[244] S. Ziyabari *et al.*, "Objective evaluation metrics for automatic classification of EEG events", *arXiv:1712.10107*, 2019.

[245] V. Shah *et al.*, "Validation of Temporal Scoring Metrics for Automatic Seizure Detection", in *SPMB*, 2020. DOI: 10.1109/SPMB50085.2020.9353631.

[246] S. N. Baldassano *et al.*, "Crowdsourcing seizure detection: algorithm development and validation on human implanted device recordings", *Brain: A Journal of Neurology*, vol. 140, no. 6, pp. 1680–1691, 2017. DOI: 10.1093/brain/awx098.

[247] A. K. Jaiswal and H. Banka, "Local pattern transformation based feature extraction techniques for classification of epileptic EEG signals", *Biomedical Signal Processing and Control*, vol. 34, pp. 81–92, 2017. DOI: 10.1016/j.bspc.2017.01.005.

[248] W. Zhou *et al.*, "Epileptic seizure detection using lacunarity and bayesian linear discriminant analysis in intracranial EEG", *IEEE transactions on bio-medical engineering*, vol. 60, no. 12, pp. 3375–3381, 2013. DOI: 10.1109/TBME.2013.2254486.

[249] R. Hussein *et al.*, "Robust detection of epileptic seizures using deep neural networks", in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2546–2550. DOI: 10.1109/ICASSP.2018.8462029.

[250] S. Beniczky and P. Ryvlin, "Standards for testing and clinical validation of seizure detection devices", *Epilepsia*, vol. 59, pp. 9–13, 2018. DOI: 10.1111/epi.14049.

[251] M. Galar *et al.*, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 42, no. 4, pp. 463–484, 2012. DOI: 10.1109/TSMCC.2011.2161285.

[252] F. Yang *et al.*, "Using random forest for reliable classification and cost-sensitive learning for medical diagnosis", *BMC Bioinformatics*, vol. 10, 2009. DOI: 10.1186/1471-2105-10-S1-S22.

[253] D. F. Ghougassian *et al.*, "Evaluating the utility of inpatient video-EEG monitoring", *Epilepsia*, vol. 45, no. 8, pp. 928–932, 2004. DOI: 10.1111/j.0013-9580.2004.51003.x.

[254] D. Schmidt and M. Sillanpää, "Evidence-based review on the natural history of the epilepsies", *Current Opinion in Neurology*, vol. 25, no. 2, pp. 159–163, 2012. DOI: 10.1097/WCO.0b013e3283507e73.

[255] S. Supriya *et al.*, "Automated epilepsy detection techniques from electroencephalogram signals: a review study", *Health Information Science and Systems*, vol. 8, no. 1, p. 33, 2020. DOI: 10.1007/s13755-020-00129-1.

[256] S. Wong *et al.*, "EEG datasets for seizure detection and prediction— a review", *Epilepsia Open*, vol. n/a, n/a 2023. DOI: 10.1002/epi4.12704.

[257] P. Handa *et al.*, "Open and free EEG datasets for epilepsy diagnosis", *arXiv:2108.01030*, 2021.

[258] R. G. Andrzejak *et al.*, "Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: dependence on recording region and brain state", *Physical Review E*, vol. 64, no. 6, p. 061 907, 2001. DOI: 10.1103/PhysRevE.64.061907.

[259] V. Shah *et al.*, "The temple university hospital seizure detection corpus", *Frontiers in Neuroinformatics*, vol. 12, 2018.

[260] "UPenn and mayo clinic's seizure detection challenge". (), [Online]. Available: https://kaggle.com/competitions/seizure-detection (visited on 02/24/2023).

[261] M. Bhagubai. "BIOMED seizure detection challenge". (), [Online]. Available: https://biomedepi.github.io/seizure_detection_challenge/ (visited on 05/23/2023).

[262] "My seizure gauge", Epilepsy Ecosystem. (), [Online]. Available: https://www.epilepsyecosystem.org/my-seizure-gauge-1 (visited on 05/23/2023).

[263] W. Nasreddine, "Epileptic EEG dataset", 2021. DOI: 10.17632/5pc2j46cbc.1.

[264] "American epilepsy society seizure prediction challenge". (), [Online]. Available: https://kaggle.com/competitions/seizure-prediction (visited on 02/24/2023).

[265] P. Detti, *Siena scalp EEG database*. DOI: 10.13026/5D4A-J060.

[266] "Melbourne NeuroVista seizure prediction trial", 2018. DOI: 10.26188/5b6a999fa2316.

[267] P. Swami *et al.*, *EEG epilepsy datasets*, 2016. DOI: 10.13140/RG.2.2.14280.32006.

[268] S. Panwar, *Single electrode EEG data of healthy and epileptic patients*, 2020. DOI: 10.5281/zenodo.3684992.

[269] A. Burrello *et al.*, "Laelaps: An Energy-Efficient Seizure Detection Algorithm from Long-term Human iEEG Recordings without False Alarms", *Design Automation and Test in Europe (DATE)*, 2019. DOI: 10.23919/DATE.2019.8715186.

[270] N. Stevenson *et al.*, *A dataset of neonatal EEG recordings with seizures annotations*, 2018. DOI: 10.5281/zenodo.2547147.

[271] D. Cserpan *et al.*, *Dataset of EEG recordings of pediatric patients with epilepsy based on the 10-20 system*, 2021. DOI: 10.18112/OPENNEURO.DS003555.V1.0.1.

[272] S. Rheims *et al.*, "Hypoxemia following generalized convulsive seizures: Risk factors and effect of oxygen therapy", *Neurology*, vol. 92, e183–e193, 2019. DOI: 10.1212/WNL.0000000000006777.

[273] M. K. Smyk and G. van Luijtelaar, "Circadian rhythms and epilepsy: a suitable case for absence epilepsy", *Frontiers in Neurology*, vol. 11, 2020.

[274] C. T. Anderson *et al.*, "Day-night patterns of epileptiform activity in 65 patients with long-term ambulatory electrocorticography", *Journal of Clinical Neurophysiology*, vol. 32, no. 5, pp. 406–412, 2015. DOI: 10.1097/WNP.0000000000000183.

[275] M. J. Cook *et al.*, "The dynamics of the epileptic brain reveal long-memory processes", *Frontiers in Neurology*, vol. 5, p. 217, 2014. DOI: 10.3389/fneur.2014.00217.

[276] V. P. Kumaravel *et al.*, *Knowledge distillation-based channel reduction for wearable EEG applications*, 2023. DOI: 10.36227/techrxiv.22651156.v1.

[277] F. Lopes *et al.*, "Removing artefacts and periodically retraining improve performance of neural network-based seizure prediction models", *Scientific Reports*, vol. 13, no. 1, p. 5918, 2023. DOI: 10.1038/s41598-023-30864-w.

[278] M. K. Islam *et al.*, "A wavelet-based artifact reduction from scalp EEG for epileptic seizure detection", *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 5, pp. 1321–1332, 2016. DOI: 10.1109/JBHI.2015.2457093.

[279] G. H. Klem *et al.*, "The ten-twenty electrode system of the International Federation. The International Federation of Clinical Neurophysiology.", *Electroencephalogr. clin. neurophysiol.*, vol. 52, pp. 3–6, 1999.

[280] V. Gabeff *et al.*, "Interpreting deep learning models for epileptic seizure detection on EEG signals", *Artificial Intelligence in Medicine*, vol. 117, p. 102 084, 2021. DOI: 10.1016/j.artmed.2021.102084.

[281] B. Hjorth, "EEG analysis based on time domain properties", *Electroencephalography and Clinical Neurophysiology*, vol. 29, no. 3, pp. 306–310, 1970. DOI: 10.1016/0013-4694(70)90143-4.

[282] P. Perucca *et al.*, "Widespread EEG changes precede focal seizures", *PLOS ONE*, vol. 8, no. 11, e80972, 2013. DOI: 10.1371/journal.pone.0080972.

[283] M. Heers *et al.*, "Spectral bandwidth of interictal fast epileptic activity characterizes the seizure onset zone", *NeuroImage: Clinical*, vol. 17, pp. 865–872, 2018. DOI: 10.1016/j.nicl.2017.11.021.

[284] I. Stancin *et al.*, "A review of EEG signal features and their application in driver drowsiness detection systems", *Sensors*, vol. 21, no. 11, p. 3786, 2021. DOI: 10.3390/s21113786.

[285] I. I. Shevchenko, "Lyapunov exponents in resonance multiplets", *Physics Letters A*, vol. 378, no. 1, pp. 34–42, 2014. DOI: 10.1016/j.physleta.2013.10.035.

[286] A. Lempel and J. Ziv, "On the complexity of finite sequences", *IEEE Transactions on Information Theory*, vol. 22, no. 1, pp. 75–81, 1976. DOI: 10.1109/TIT.1976.1055501.

[287] M. Cohen *et al.*, "Applying continuous chaotic modeling to cardiac signal analysis", *IEEE Engineering in Medicine and Biology Magazine*, vol. 15, no. 5, pp. 97–102, 1996. DOI: 10.1109/51.537065.

[288] D. Abásolo *et al.*, "Approximate entropy and auto mutual information analysis of the electroencephalogram in alzheimer's disease patients", *Medical & Biological Engineering & Computing*, vol. 46, no. 10, pp. 1019–1028, 2008. DOI: 10.1007/s11517-008-0392-1.

[289] C. E. Shannon, "A mathematical theory of communication", *The Bell System Technical Journal*, vol. 27, no. 4, pp. 623–656, 1948. DOI: 10.1002/j.1538-7305.1948.tb00917.x.

[290] A. Renyi, *Probability Theory*. 2007.

[291] C. Tsallis *et al.*, "The role of constraints within generalized nonextensive statistics", *Physica A: Statistical Mechanics and its Applications*, vol. 261, no. 3, pp. 534–554, 1998. DOI: 10.1016/S0378-4371(98)00437-3.

[292] S. M. Pincus *et al.*, "A regularity statistic for medical data analysis", *Journal of Clinical Monitoring*, vol. 7, no. 4, pp. 335–345, 1991. DOI: 10.1007/BF01619355.

[293] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy", *American Journal of Physiology. Heart and Circulatory Physiology*, vol. 278, no. 6, H2039–2049, 2000. DOI: 10.1152/ajpheart.2000.278.6.H2039.

[294] D. Cuesta-Frau *et al.*, "Embedded dimension and time series length. practical influence on permutation entropy and its applications", *Entropy*, vol. 21, no. 4, p. 385, 2019. DOI: 10.3390/e21040385.

[295] E. Pereda *et al.*, "Nonlinear multivariate analysis of neurophysiological signals", *Progress in Neurobiology*, vol. 77, no. 1, pp. 1–37, 2005. DOI: 10.1016/j.pneurobio.2005.10.003.

[296] P. H. Randolph, "Spectral analysis and its applications", *Technometrics*, vol. 12, no. 1, pp. 174–175, 1970. DOI: 10.1080/00401706.1970.10488651.

[297] C. W. J. Granger, "Investigating causal relations by econometric models and cross-spectral methods", *Econometrica*, vol. 37, no. 3, pp. 424–438, 1969. DOI: 10.2307/1912791.

[298] H. Ocak, "Automatic detection of epileptic seizures in EEG using discrete wavelet transform and approximate entropy", *Expert Systems with Applications*, vol. 36, no. 2, pp. 2027–2036, 2009. DOI: 10.1016/j.eswa.2007.12.065.

[299] L. S. Vidyaratne and K. M. Iftekharuddin, "Real-time epileptic seizure detection using EEG", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 11, pp. 2146–2156, 2017. DOI: 10.1109/TNSRE.2017.2697920.

[300] A. Page *et al.*, "Wearable seizure detection using convolutional neural networks with transfer learning", in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2016, pp. 1086–1089. DOI: 10.1109/ISCAS.2016.7527433.

[301] A. Mohammed *et al.*, "Time-series cross-validation parallel programming using mpi", *International Conference on Data Analytics for Business and Industry (ICDABI)*, 2021.

[302] R. Zanetti *et al.*, "Approximate zero-crossing: a new interpretable, highly discriminative and low-complexity feature for EEG and iEEG seizure detection", *Journal of Neural Engineering*, vol. 19, no. 6, 2022. DOI: 10.1088/1741-2552/aca1e4.

[303] N. Japkowicz and M. Shah, *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge Uni Press, 2011. DOI: 10.1017/CBO9780511921803.

[304] S. B. Wilson *et al.*, "Seizure detection: correlation of human experts", *Clinical Neurophysiology*, vol. 114, no. 11, pp. 2156–2164, 2003. DOI: 10.1016/s1388-2457(03)00212-8.

[305] U. Pale *et al.*, "Hyperdimensional computing encoding for feature selection on the use case of epileptic seizure detection", *arXiv:2205.07654*, 2022. DOI: 10.48550/arXiv.2205.07654.

[306] R. Zanetti *et al.*, "Real-Time EEG-Based Cognitive Workload Monitoring on Wearable Devices", *IEEE transactions on bio-medical engineering*, vol. 69, no. 1, pp. 265–277, 2022. DOI: 10.1109/TBME.2021.3092206.

[307] S. Wang *et al.*, "Deep learning for spatio-temporal data mining: a survey", *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 8, pp. 3681–3700, 2022. DOI: 10.1109/TKDE.2020.3025580.

[308] T. Morita *et al.*, "Contribution of neuroimaging studies to understanding development of human cognitive brain functions", *Frontiers in Human Neuroscience*, vol. 10, p. 464, 2016. DOI: 10.3389/fnhum.2016.00464.

[309] C. Martinelli and S. S. Shergill, "Everything you wanted to know about neuroimaging and psychiatry, but were afraid to ask", *BJPsych Advances*, vol. 21, no. 4, pp. 251–260, 2015. DOI: 10.1192/apt.bp.114.013763.

[310] Z. Wu *et al.*, *A Comprehensive Survey on Graph Neural Networks*. 2019.

[311] I. Roesch and T. Günther, "Visualization of neural network predictions for weather forecasting", *Computer Graphics Forum*, vol. 38, no. 1, pp. 209–220, 2019. DOI: 10.1111/cgf.13453.

[312] Y. Kaya *et al.*, "1D-local binary pattern based feature extraction for classification of epileptic EEG signals", *Applied Mathematics and Computation*, vol. 243, pp. 209–219, 2014. DOI: 10.1016/j.amc.2014.05.128.

[313] T. M. Ingolfsson *et al.*, "Towards long-term non-invasive monitoring for epilepsy via wearable EEG devices", in *IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2021, pp. 01–04. DOI: 10.1109/BioCAS49922.2021.9644949.

[314] E. Bruno *et al.*, "Wearable technology in epilepsy: the views of patients, caregivers, and healthcare professionals", *Epilepsy & Behavior*, vol. 85, 2018. DOI: 10.1016/j.yebeh.2018.05.044.

[315] M. Heddes *et al.*, *Torchhd: an open-source python library to support hyperdimensional computing research*, 2022. DOI: 10.48550/arXiv.2205.09208.

[316] I. Nunes *et al.*, *An extension to basis-hypervectors for learning from circular data in hyperdimensional computing*, 2022. DOI: 10.48550/arXiv.2205.07920.

[317] M. Teplan, "Fundamental of EEG Measurement", *Meas. Sci. Rev.*, vol. 2, 2002.

[318] F. Forooghifar *et al.*, "Self-aware wearable systems in epileptic seizure detection", in *Euromicro Conference on Digital System Design*, 2018, pp. 426–432. DOI: 10.1109/DSD.2018.00078.

[319] L. Ferretti *et al.*, "INCLASS: incremental classification strategy for self-aware epileptic seizure detection", in *Design, Automation & Test in Europe Conference & Exhibition*, 2022, pp. 1449–1454. DOI: 10.23919/DATE54114.2022.9774713.

[320] G. Bhanot *et al.*, "Biomedical data analysis in translational research: integration of expert knowledge and interpretable models", in *The European Symposium on Artificial Neural Networks*, 2017.

[321] A. Bhattacharyya and R. B. Pachori, "A multivariate approach for patient-specific EEG seizure detection using empirical wavelet transform", *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 9, pp. 2003–2015, 2017. DOI: 10.1109/TBME.2017.2650259.

[322] E. Keogh *et al.*, "Segmenting time series: a survey and novel approach", *Data Mining in Time Series Databases*, vol. 57, 2003. DOI: 10.1142/9789812565402_0001.

[323] S. Zanoli *et al.*, "An error-based approximation sensing circuit for event-triggered, low power wearable sensors", in *ArXiv*. DOI: 10.48550/arXiv.2106.13545.

[324] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature", *Cartographica: The International Journal for Geographic Information and Geovisualization*, vol. 10, no. 2, pp. 112–122, 1973. DOI: 10.3138/FM57-6770-U75U-7727.

[325] D. Sopic *et al.*, "Personalized seizure signature: an interpretable approach to false alarm reduction for long-term epileptic seizure detection", *Epilepsia*, 2022. DOI: 10.1111/epi.17176.

[326] U. R. Acharya *et al.*, "Application of entropies for automated diagnosis of epilepsy using EEG signals: a review", *Knowledge-Based Systems*, vol. 88, pp. 85–96, 2015. DOI: 10.1016/j.knosys.2015.08.004.

[327] F. Perez-Cruz, "Kullback-leibler divergence estimation of continuous distributions", in *IEEE International Symposium on Information Theory*, 2008. DOI: 10.1109/ISIT.2008.4595271.

[328] S. Smith, "EEG in the diagnosis, classification, and management of patients with epilepsy", *Journal of Neurology, Neurosurgery, and Psychiatry*, vol. 76, pp. ii2–ii7, 2005. DOI: 10.1136/jnnp.2005.069245.

[329] C. Gómez *et al.*, "Automatic seizure detection based on imaged-EEG signals through fully convolutional networks", *Scientific Reports*, vol. 10, no. 1, p. 21 833, 2020. DOI: 10.1038/s41598-020-78784-3.

[330] A. Rahimi *et al.*, "Hyperdimensional biosignal processing: A case study for EMG-based hand gesture recognition", in *2016 IEEE International Conference on Rebooting Computing (ICRC)*, 2016, pp. 1–8. DOI: 10.1109/ICRC.2016.7738683.

[331] A. Moin *et al.*, "Adaptive EMG-based hand gesture recognition using hyperdimensional computing", *ArXiv*, 2019.

[332] G. Karunaratne *et al.*, "Energy Efficient In-Memory Hyperdimensional Encoding for Spatio-Temporal Signal Processing", *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 5, 2021. DOI: 10.1109/TCSII.2021.3068126.

[333] L. Ge and K. K. Parhi, "Seizure Detection Using Power Spectral Density via Hyperdimensional Computing", in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021. DOI: 10.1109/ICASSP39728.2021.9414083.

[334] I. Nunes *et al.*, *GraphHD: efficient graph classification using hyperdimensional computing*, 2022. DOI: 10.48550/arXiv.2205.07826.

[335] A. Hernández-Cano *et al.*, "A Framework for Efficient and Binary Clustering in High-Dimensional Space", in *Conference Exhibition on Design, Automation Test in Europe (DATE)*, 2021. DOI: 10.23919/DATE51398.2021.9474008.

[336] N. Japkowicz and S. Stephen, "The class imbalance problem: a systematic study", *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429–449, 2002.

[337] A. L'Heureux *et al.*, "Machine learning with big data: challenges and approaches", *IEEE Access*, vol. 5, pp. 7776–7797, 2017. DOI: 10.1109/ACCESS.2017.2696365.

[338] H. E. Saadi *et al.*, "Informed under-sampling for enhancing patient specific epileptic seizure detection", *International Journal of Computer Applications*, vol. 57, no. 16, pp. 41–46, 2012.

[339] S. Amin and A. M. Kamboh, "A robust approach towards epileptic seizure detection", in *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2016, pp. 1–6. DOI: 10.1109/MLSP.2016.7738825.

[340] D. Chen *et al.*, "A high-performance seizure detection algorithm based on discrete wavelet transform (DWT) and EEG", *PLOS ONE*, vol. 12, no. 3, 2017. DOI: 10.1371/journal.pone.0173138.

[341] C. Ling and V. Sheng, "Cost-sensitive learning and the class imbalance problem", *Encyclopedia of Machine Learning*, 2010.

[342] J. Birjandtalab *et al.*, "Imbalance learning using neural networks for seizure detection", in *IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2018, pp. 1–4. DOI: 10.1109/BIOCAS.2018.8584683.

[343] Q. Yuan *et al.*, "Epileptic seizure detection based on imbalanced classification and wavelet packet transform", *Seizure*, vol. 50, pp. 99–108, 2017. DOI: 10.1016/j.seizure.2017.05.018.

[344] "UCI machine learning repository: ISOLET data set". (), [Online]. Available: http://archive.ics.uci.edu/ml/datasets/ISOLET (visited on 03/20/2023).

[345] "UCI machine learning repository: human activity recognition using smartphones data set". (), [Online]. Available: https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones (visited on 03/20/2023).

[346] G. Griffin *et al.* "Caltech-256 object category dataset". (), [Online]. Available: https://resolver.caltech.edu/CaltechAUTHORS:CNS-TR-2007-001.

[347] "UCI machine learning repository: cardiotocography data set". (), [Online]. Available: https://archive.ics.uci.edu/ml/datasets/cardiotocography (visited on 03/20/2023).

[348] M. Imani *et al.*, "HDCluster: An Accurate Clustering Using Brain-Inspired High-Dimensional Computing", in *Conference Exhibition on Design, Automation Test in Europe (DATE)*, 2019, pp. 1591–1594. DOI: 10.23919/DATE.2019.8715147.

[349] A. Hernández-Cano *et al.*, "Real-Time and Robust Hyperdimensional Classification", in *Great Lakes Symposium on VLSI (GLSVLSI)*, 2021, pp. 397–402. DOI: 10.1145/3453688.3461749.

[350] U. Pale *et al.*, "Systematic Assessment of Hyperdimensional Computing for Epileptic Seizure Detection", *International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2021. DOI: 10.1109/EMBC46164.2021.9629648.

[351] P. Kanerva, "Hyperdimensional computing: an introduction to computing in distributed representation with high-dimensional random vectors", *Cognitive computation*, 2009.

[352] T. F. Wu *et al.*, "Brain-inspired computing exploiting carbon nanotube FETs and resistive RAM: hyperdimensional computing case study", *IEEE International Solid-State Circuits Conference (ISSCC)*, 2018.

[353] S. Gupta *et al.*, "FELIX: Fast and Energy-Efficient Logic in Memory", in *IEEE International Conference on Computer-Aided Design (ICCAD)*, 2018, pp. 1–7. DOI: 10.1145/3240765.3240811.

[354] G. Karunaratne *et al.*, "In-memory hyperdimensional computing", *Nature Electronics*, vol. 3, no. 6, pp. 327–337, 2020. DOI: 10.1038/s41928-020-0410-3.

[355] H. He, "The state of machine learning frameworks in 2019", *The Gradient*, 2019.

[356] D. Dua and C. Graff, *UCI machine learning repository*, 2017.

[357] V. Ferastraoaru *et al.*, "Characteristics of large patient-reported outcomes: where can one million seizures get us?", *Epilepsia Open*, vol. 3, no. 3, pp. 364–373, 2018. DOI: 10.1002/epi4.12237.

[358] S. S. Spencer *et al.*, "Ictal effects of anticonvulsant medication withdrawal in epileptic patients", *Epilepsia*, vol. 22, no. 3, pp. 297–307, 1981. DOI: 10.1111/j.1528-1157.1981.tb04113.x.

[359] G. Alarcon *et al.*, "Power spectrum and intracranial EEG patterns at seizure onset in partial epilepsy", *Electroencephalography and Clinical Neurophysiology*, vol. 94, no. 5, pp. 326–337, DOI: 10.1016/0013-4694(94)00286-t.

[360] J. S. Naftulin *et al.*, "Ictal and preictal power changes outside of the seizure focus correlate with seizure generalization", *Epilepsia*, vol. 59, no. 7, pp. 1398–1409, 2018. DOI: 10.1111/epi.14449.

[361] L.-E. Martinet *et al.*, "Slow spatial recruitment of neocortex during secondarily generalized seizures and its relation to surgical outcome", *The Journal of Neuroscience*, vol. 35, no. 25, pp. 9477–9490, 2015. DOI: 10.1523/JNEUROSCI.0049-15.2015.

[362] L. Wu and J. Gotman, "Segmentation and classification of EEG during epileptic seizures", *Electroencephalography and Clinical Neurophysiology*, vol. 106, no. 4, pp. 344–356, 1998. DOI: 10.1016/s0013-4694(97)00156-9.

[363] F. Wendling *et al.*, "Extraction of spatio-temporal signatures from depth EEG seizure signals based on objective matching in warped vectorial observations", *IEEE transactions on biomedical engineering*, vol. 43, no. 10, pp. 990–1000, 1996. DOI: 10.1109/10.536900.

[364] V. Louis Dorr *et al.*, "Extraction of reproducible seizure patterns based on EEG scalp correlations", *Biomedical Signal Processing and Control*, IFAC Symposia on Biomedical Systems Modelling & Control, vol. 2, no. 3, pp. 154–162, 2007. DOI: 10.1016/j.bspc.2007.07.002.

[365] G. M. Schroeder *et al.*, "Seizure pathways change on circadian and slower timescales in individual patients with focal epilepsy", *Proceedings of the National Academy of Sciences of the United States of America*, vol. 117, no. 20, pp. 11 048–11 058, 2020. DOI: 10.1073/pnas.1922084117.

[366] R. J. McGinn *et al.*, "Precision medicine in epilepsy", *Progress in Molecular Biology and Translational Science*, vol. 190, no. 1, pp. 147–188, 2022. DOI: 10.1016/bs.pmbts.2022.04.001.

[367] W. A. Simon *et al.*, "HDTorch: Accelerating Hyperdimensional Computing with GP-GPUs for Design Space Exploration", in *IEEE International Conference on Computer-Aided Design (ICCAD)*, 2022, pp. 1–8. DOI: 10.1145/3508352.3549475.

[368] U. Pale *et al.*, *Importance of methodological choices in data manipulation for validating epileptic seizure detection models*, 2023. DOI: 10.48550/arXiv.2302.10672.

[369] A. Holzinger *et al.*, "What do we need to build explainable AI systems for the medical domain?", *ArXiv*, 2017.

[370] M. A. Ahmad *et al.*, "Interpretable machine learning in healthcare", in *IEEE International Conference on Healthcare Informatics (ICHI)*, 2018, pp. 447–447. DOI: 10.1109/ICHI.2018.00095.

[371] W. J. Murdoch *et al.*, "Definitions, methods, and applications in interpretable machine learning", *Proceedings of the National Academy of Sciences*, vol. 116, no. 44, pp. 22 071–22 080, 2019. DOI: 10.1073/pnas.1900654116.

[372] A. Chaddad *et al.*, "Survey of explainable AI techniques in healthcare", *Sensors*, vol. 23, no. 2, p. 634, 2023. DOI: 10.3390/s23020634.

[373] T. A. A. Abdullah *et al.*, "A review of interpretable ML in healthcare: taxonomy, applications, challenges, and future directions", *Symmetry*, vol. 13, no. 12, p. 2439, 2021. DOI: 10.3390/sym13122439.

[374] R. Guidotti *et al.*, "A survey of methods for explaining black box models", *ACM Computing Surveys*, vol. 51, no. 5, 93:1–93:42, 2018. DOI: 10.1145/3236009.

[375] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead", *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019. DOI: 10.1038/s42256-019-0048-x.

[376] A. Esteva *et al.*, "A guide to deep learning in healthcare", *Nature Medicine*, vol. 25, no. 1, pp. 24–29, 2019. DOI: 10.1038/s41591-018-0316-z.

[377] A. Vellido *et al.*, *Seeing is believing: The importance of visualization in real-world machine learning applications*. 2011.

[378] S. Liu *et al.*, "Towards better analysis of machine learning models: a visual analytics perspective", *Visual Informatics*, vol. 1, no. 1, pp. 48–56, 2017. DOI: 10.1016/j.visinf.2017.01.006.

[379] D. A. Keim *et al.*, "Visual analytics: scope and challenges", in *Visual Data Mining: Theory, Techniques and Tools for Visual Analytics*, ser. Lecture Notes in Computer Science, S. J. Simoff *et al.*, Eds., Springer, 2008, pp. 76–90.

[380] A. Rahimi *et al.*, *Hyperdimensional Computing for Noninvasive Brain–Computer Interfaces: Blind and One-Shot Classification of EEG Error-Related Potentials*. 2017.

[381] H. Daoud and M. Bayoumi, "Deep learning approach for epileptic focus localization", *IEEE Transactions on Biomedical Circuits and Systems*, 2020. DOI: 10.1109/TBCAS.2019.2957087.

[382] J. Craley *et al.*, "Automated seizure activity tracking and onset zone localization from scalp EEG using deep neural networks", *PLOS ONE*, vol. 17, no. 2, 2022. DOI: 10.1371/journal.pone.0264537.

[383] J. Birjandtalab *et al.*, "Automated seizure detection using limited-channel EEG and non-linear dimension reduction", *Computers in Biology and Medicine*, vol. 82, pp. 49–58, 2017. DOI: 10.1016/j.compbiomed.2017.01.011.

[384] H. Daoud and M. Bayoumi, "Efficient epileptic seizure prediction based on deep learning", *IEEE Transactions on Biomedical Circuits and Systems*, vol. 13, no. 5, 2019. DOI: 10.1109/TBCAS.2019.2929053.

[385] M. R. Karimi and H. Kassiri, "A multi-feature nonlinear-SVM seizure detection algorithm with patient-specific channel selection and feature customization", in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2020. DOI: 10.1109/ISCAS45731.2020.9180729.

[386] A. A. Ein Shoka *et al.*, "Automated seizure diagnosis system based on feature extraction and channel selection using EEG signals", *Brain Informatics*, vol. 8, no. 1, p. 1, 2021. DOI: 10.1186/s40708-021-00123-7.

[387] S. Chakrabarti *et al.*, "A channel selection method for epileptic EEG signals", in *Emerging Technologies in Data Mining and Information Security*, A. Abraham *et al.*, Eds., Springer, 2019, pp. 565–573. DOI: 10.1007/978-981-13-1951-8_51.

[388] R. Jana and I. Mukherjee, "Deep learning based efficient epileptic seizure prediction with EEG channel optimization", *Biomedical Signal Processing and Control*, vol. 68, p. 102 767, 2021. DOI: 10.1016/j.bspc.2021.102767.

[389] V. Shah *et al.*, "Optimizing channel selection for seizure detection", in *IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, 2017, pp. 1–5. DOI: 10.1109/SPMB.2017.8257019.

[390] A. Affes *et al.*, "Personalized attention-based EEG channel selection for epileptic seizure prediction", *Expert Systems with Applications*, vol. 206, 2022. DOI: 10.1016/j.eswa.2022.117733.

[391] M. Hersche *et al.*, *Constrained few-shot class-incremental learning*, 2022. DOI: 10.48550/arXiv.2203.16588.

UNA PALE

"Electrical engineer doing data science for biomedical applications, passionate about real-life problems and projects that make a difference."

Address:	**Lausanne, Switzerland**
Telephone:	**+041 77 9130 182, +385 98 186 2526**
E-mail:	**una.pale@gmail.com, una.pale@epfl.ch**
Web page:	**www.una-pale.from.hr**
Date and place of birth:	**1992, Croatia**

ACADEMIC EDUCATION

- 02/2018 – Present **PhD student in electrical engineering/neuroscience**
  École polytechnique fédérale de Lausanne, Switzerland
- 09/2014 – 09/2016 **Master of Science in Electrical Engineering and Information Technology**
  Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia (GPA: 4.8/ 5.0)
- 09/2015 – 03/2016 **Erasmus exchange**
  at Master program Biomedical Engineering, Technical University of Vienna, Austria
- 09/2011 – 07/2014 **Bachelor of Science in Electrical Engineering and Information Technology**
  Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia (GPA: 4.7/ 5.0)

WORK EXPERIENCE

- 12/2018 – Present **NGO Penkala – Croatian Association of Young Scientists**
  Board member, project leader and manager
  Leading the association, funding applications, deciding on the main plans, activities and future directions. Working on building Croatian scientific network and providing additional education for Croatian scientists.
- 02/2017 – 06/2017 **Development engineer**
  UMO neuroscience j.d.o.o, Zagreb, Croatia
  Working on a patent of an algorithm for automatic detection of Cz electrode on a neurofeedback headset which was developed within the company.
- 07/2015 – 09/2015 **Summer intern**
  Robert Bosch GmbH, Robert Bosch Research Center Renningen, Stuttgart, Germany
  Programming Fanuc robot and industrial cameras for automatic video inspection of products.

PROJECTS

Non-scientific projects:

- "Science and Policy Summer School", 2022 - Present
  - Initiator and coorganizer of PhD summer school funded by EPFL/ETHZ, https://scienceandpolicy2023.epfl.ch/
- "Istražilica" istrazilica.hr, 2022 - Present
  - Initiator and working on creating Web platform for scientific opportunities for young Croatian scientists, project in the scope of NFO Penkala
- Map of Croatian Scientists, 2022 - Present
  - Initiator and working on creating Web platform with a map of Croatian scientists, project in the scope of NGO Penkala
- Podcast "Slušalica", 2018 - Present
  - Starting the first Croatian scientific podcast, host of many episodes
- International competition IYNT, 2014-2018
  - Coorganizing Croatian selection for the International Young Naturalists' Tournament, helping students in solving scientific problems and preparation for the competition
- "Youth Research Center" NGO, 2012 - 2016
  - Leading student organization, ensuring funding, organizing workshops and competitions

Scientific projects:

- Project 09/2016 – 08/2017
  - *Nadi shodana breathing techinique's influence on autonomic nervous system*

Research on how does yoga breathing technique "Nadi shodana" affects on autonomic nervous, conducted at the University Hospital Centre in Zagreb, Croatia.

- MSc thesis
  03/2016 – 09/2016
  - *Electromyographic biofeedback system*

  Design, construction and testing of a compact-size, wireless (BLE) and surface EMG measurement system for physiotherapy, with developed Android application (Java) for user interface.
- Graduate project
  09/2015 – 02/2016
  - *Contactless assessment of HR using Eulerian video magnification*

  Eulerian video magnification applied to video recordings of palpation sites of human for contactless heart rate (HR) and pulse transit time (PTT) assessment. Project was done at Technical University of Vienna, Austria.
- Project
  09/2015 – 11/2016
  - *Heart rate variability analysis using wavelet transform*

  Assessment of three methods of wavelet transform for heart rate variability (HRV) extraction from HR signal. Project was done at Technical University of Vienna, Austria.
- BSc thesis
  03/2014 – 07/2014
  - *Modeling of bee's movements in a biohybrid simulator*

  Developing two models of bees' behaviour and movement in a field with a temperature gradient.
- Project
  10/2013 – 02/2014
  - *Coordinated control of autonomous quadcopter and mobile robots*

  Drone image processing for detection and localization of mobile robots in arena for rescue missions.

SCIENTIFIC PAPERS AND CONFERENCES

- U. Pale, T. Teijeiro, S. Rheims, P. Ryvlin, and D. Atienza, "Combining General and Personalized Models for Epilepsy Detection with Hyperdimensional Computing", arXiv
- U. Pale, T. Teijeiro, and D. Atienza, "Importance of methodological choices in data manipulation for validating epileptic seizure detection models", arXiv:2302.10672
- U. Pale, T. Teijeiro, and D. Atienza, "ExG Signal Feature Selection Using Hyperdimensional Computing Encoding", BIBM 2022
- R. Zanetti, U. Pale, T. Teijeiro, and D. Atienza, "Approximate zero-crossing: a new interpretable, highly discriminative and low-complexity feature for EEG and iEEG seizure detection", Journal of Neural Engineering, 2022
- W.Simon, U.Pale, T. Teijeiro, and D. Atienza, "HDTorch: Accelerating Hyperdimensional Computing with GP-GPUs for Design Space Exploration", ICCAD 2022
- U. Pale, T. Teijeiro, and D. Atienza, "Exploration of Hyperdimensional Computing Strategies for Enhanced Learning on Epileptic Seizure Detection", EMBC 2022
- U. Pale, T. Teijeiro, and D. Atienza, "Multi-Centroid Hyperdimensional Computing Approach for Epileptic Seizure Detection", Frontiers in Neurology
- U. Pale, T. Teijeiro, and D. Atienza, "Systematic Assessment of Hyperdimensional Computing for Epileptic Seizure Detection," EMBC 2021
- F. Dell'Agnola, U. Pale, R. Marino, A. Arza Valdes, and D. Atienza, "MBioTracker: Multimodal Self-Aware Bio-Monitoring Wearable System for Online Workload Detection," IEEE Trans. Biomed. Circuits Syst., 2021
- U. Pale, N. Müller, A. Arza, and D. Atienza, "ReBeatICG: Real-time Low-Complexity Beat-to-beat Impedance Cardiogram Delineation Algorithm," ArXiv210501525, 2021
- Pale U., Atzori M., Müller H., Scano A.: Synergies in Hand Grasps: Analysis of Intra- and Inter-Session Data, MDPI Sensors 2020
- Pale U., Cifrek M., Krois I., Peharec S.: Personal electromyographic biofeedback system „MyMyo" CMBEBIH International Conference on Medical and Biological Engineering, 2017

HONOURS AND AWARDS

- Gold medal in 9th *"International Exhibition of Inventions"* (Kunshan, China) 2016 and Silver medal in 14th *"International Innovation Exhibition"* (Zagreb, Croatia) 2016 with „Personal electromyographic biofeedback system - MyMyo"
- *Rector's Award, 2015* - for work on project titled "Audio phonebook for the blind people"

- BEST Engineering Competition (Team Design category), 2013 - European competition in designing and constructing a prototype which fulfills task's requirements within a limited amount of time and material. Won 6. place in finals.
- *Dean's Award "Josip Lončar"*, 2012 - for outstanding performance in the 2nd year of undergraduate, University of Zagreb
- Scholarships
  - The *"City of Zagreb"* University Scholarship (2013 – 2016)
  - *"Internship Programme of German Business for the Countries of the Western Balkans* foundation scholarship (2015)
  - *DAAD* scholarship for German language summer course (2014)
  - The *"City of Zagreb"* High School Scholarship (2009 – 2013)
- International Young Physicists Tournament (IYPT), Teheran, Iran, 2011, bronze medal
- International Young Physicists Tournament (IYPT), Vienna, Austria, 2010, bronze medal

TEACHING EXPERIENCE

• 02/2018 – present	Mentoring bachelor and master students as well as teaching assistant at the EPFL, Lausanne
• 07/2013	Workshop leader at "Summer Science Factory 2013", Samobor, Croatia ("Electromagnetic Fellowship" for 7th and 8th grade of primary school)
• 09/2012 – 09/2014	Student assistant for various courses at the Faculty of electrical engineering and computing, University of Zagreb, Croatia
• 07/2012	Workshop leader at "Summer School of Science S3 + +" 2012th ("Shape memory alloy robotic arm" for students 3rd and 4th year of high school)

ADDITIONAL EDUCATION AND COURSES

• 09/2022 – 01/2023	**Innosuisse Startup Training** Bussiness Concept training at EPFL Innovation Park
• 2019 - Present	**French language course, EPFL, B1-2**
• 08/2014	**German language course, B1-2** German Courses Passau, Passau, Germany
• 09/2000 – 07/2012	**Music school** Music High School "Vatroslav Lisinski" - instrument Harp and Clarinet Music Primary School „Ivan Zajc" – instrument Piano (2000 – 2007)

PERSONAL SKILLS

Job-related skills	- versatility, persistence, attention to details, reliability and curiosity for research - ability to work in multicultural environments, challenging social situations etc. - experience as a leader in Croatian Young scientists student association "Penkala" (2018 – present) - organization skills through leading of "Youth research center" student organization (2012-2016), and Croatian selection for International Young Naturalists' Tournament (2014 – 2018) as well as many events in scope of "Penkala" - language knowledge: English C1, German B1-2, French B1
Computer skills	- good knowledge and experience with: Matlab, Python, C, Solidworks, Altium - lots of experience with various machine learning algorithms especially hyperdimensional computing - good grasp of signal and image processing algorithms, especially biomedical signals - intermediary knowledge of: Java and Android Studio, C++ - projects with microcontrollers programming and PCB design (Arduino, MSP430, Stellaris...) - basic experience in website development, podcasts recording and video processing
Hobbies	- outdoor oriented person, climbing, hiking, ski touring - had musically oriented childhood by playing piano, clarinet and harp - now more oriented on photography, painting and DIY things - recently deep dived into effective altruism