Debunking Misinformation on the Web: Detection, Validation, and Visualisation

Thèse N° 9694

Présentée le 26 juillet 2019

à la Faculté informatique et communications
Laboratoire de systèmes d'information répartis
Programme doctoral en informatique et communications

pour l'obtention du grade de Docteur ès Sciences

par

Thành Tâm NGUYÊN

Acceptée sur proposition du jury

Prof. J.-Y. Le Boudec, président du jury
Prof. K. Aberer, directeur de thèse
Dr A. Wierzbicki, rapporteur
Prof. Z. Miklos, rapporteur
Prof. B. Faltings, rapporteur

2019

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Acknowledgements

I would like to express my gratitude to many people, who generously helped my to finish the work presented in this thesis as well as continuously supported me during the course of my PhD study.

Utmost special appreciation goes to my thesis director – Prof. Karl Aberer at Distributed Information Systems Laboratory (LSIR), EPFL. Through his valuable guidance and offer of freedom, I have obtained significant skills for my future research as well as followed the right research directions. His support is unprecedented and helps me go through the most important chapter of my life. Without him, I would not be able to achieve the highest degree in academia and find the right path for my future career. Thank you so much for your understanding and bearing of me.

A profound gratitude goes to my thesis committee: Prof. Jean-Yves Le Boudec, Prof. Boi Faltings, Dr. Adam Wierzbicki, and Prof. Zoltan Miklos for their important comments and discussions to improve my dissertation. Thank you very much for your patience and your time in the organisation of my PhD exam.

It was a joyful period of my life to work with my colleagues at LSIR for their generous supports and friendships. You are my team and my closest international friends who offer me a lot of cultural and social experiences. I apologise for my work and time constraints that I am unable to follow through the team building activities.

A special thanks goes to Chantal, who helped me sort out so many not only administrative issues but also paper works. Best wishes to her and her family.

Unforgettable appreciation goes to my colleagues and my coauthors whom I collaborated with during my PhD study at EPFL, especially the master and bachelor students at LSIR.

As always, I am thankful for my parents and my friends who have understood and shared with me joys and sorrows in my life.

There are a countless number of other persons who supported me to accomplish my PhD study. I apologize for not mentioning their names here, but I will always remember their help in my heart.

Abstract

Our modern society is struggling with an unprecedented amount of online misinformation, which does harm to democracy, economics, and cybersecurity. Journalism and politics have been impacted by misinformation on a global scale, with weakened public trust in governments seen during the Brexit referendum and viral fake election stories outperforming genuine news on social media during the 2016 U.S. presidential election campaign. Online misinformation also single-handedly caused $136.5 billion in losses in the stock market value through a single tweet about explosions in the White House. Such attacks are even driven by the advances of modern artificial intelligence (AI) these days and pose a new and ever-evolving cyber threat operating at the information level, which is far more advanced than traditional cybersecurity attacks at the hardware and software levels.

Research in this area is still in its infancy but demonstrates that debunking misinformation on the Web is a formidable challenge. This is due to several reasons. First, the open nature of social platforms such as Facebook and Twitter allows users to freely produce and propagate any content without authentication, and this has been exploited to spread hundreds of thousands of fake news at a rate of more than three million social posts per minute. Second, those responsible for the spread of misinformation harvest the power of AI attacking models to mix and disguise falsehoods with common news. Methods of camouflage are used to cover digital footprints through synthesizing millions of fake accounts and appearing to participate in normal social interactions with other users. Third, innocent users, without proper alerts from algorithmic models, can accidentally spread misinformation in an exponential wave of shares, posts, and articles. The misinformation wave is often only detected when already beyond control and consequently can cause large-scale effects in a very short time.

The overarching goal of this thesis is to help media organizations, governments, the public, and academia build a misinformation debunking framework, where algorithmic models and human validators are seamlessly and cost-effectively integrated to prevent the damage of misinformation from occurring. This thesis investigates three important components of such a framework, including: (i) detection, (ii) validation, and (iii) visualisation. For each of them, we focus on a working misinformation domain that enables us to systematically design and entail consistent models and dedicated methods for solving the problem from different angles.

The main contributions of this thesis are:

- **Detection:** Early detection can potentially prevent the spread of misinformation from occurring by flagging suspicious news for human attention; however it remains, to date, an unsolved challenge. To this end, we proposed a graph-based progressive model that *detects emergent misinformation stories* from data streams of social networks. The model confirmed and leveraged the echo chamber effect of misinformation waves to identify the affected social entities via their interactions.
- **Validation:** Learning a good detection model already requires a lot of training data; and yet it can be outdated swiftly with new social trends. A promising approach is to use human experts to validate the detection results, helping algorithmic models to train themselves to become smarter and adaptive to new traits of misinformation. Realizing this approach, we proposed guidance strategies that *minimize human efforts in validating misinformation flags*, boosting the confidence of misinformation detection and reducing the risk of false alarms.
- **Visualisation:** Disseminating the debunking reports is an important step to raise public awareness against falsehood contents and educate Web users. However, human users can be easily overwhelmed by the high volume of Web data, as the level of redundancy increases and the value density decreases. To this end, we proposed a retaining protocol for streaming data to *visualise highly representative information with minimal regret*. The utility measure is designed so that the retained data covers emergent social topics, fresh social posts, and rich of social contexts.

In summary, this thesis proposed key components of building a misinformation debunking framework. The proposed techniques improve upon the state-of-the-art in a variety of misinformation domains, including rumours, Web claims, and social streams.

Keywords: *digital misinformation, anomaly detection, effort minimisation, social media analysis, streaming data visualisation*

Résumé

Notre société moderne est confrontée à un volume de mésinformation en ligne, ce qui nuit à la démocratie, l'économie et la cyber-sécurité. Le journalisme et la politique ont été impactés par la mésinformation à une échelle globale, ce qui fragilise la confiance du public envers les gouvernements. Le phénomène a récemment été exemplifié lors du referendum pour le Brexit et de la campagne présidentielle américaine de 2016 dans laquelle les "fake news" dépassaient la visibilité des articles de presse légitimes. La mésinformation en ligne a été responsable d'une perte de 136.5 milliards de dollars en valeurs boursières après un simple Tweet émanant de la maison blanche. Ce genre d'attaques profite de l'avancée de l'intelligence artificielle et constitue une nouvelle menace, en constante évolution, qui opère au niveau de l'information, rendant l'attaque bien plus complexe que lorsque la cible est un logiciel ou une machine.

La recherche visant à contrer le phénomène en est toujours à ses balbutiements mais elle suggère cependant le défi représenté par la tâche. Cela est dû à plusieurs raisons. Premièrement, la nature ouverte des plateformes telle que Facebook ou Twitter permet aux utilisateurs de partager et propager librement du contenu sans mécanismes d'authentification, ce qui a déjà été exploité pour répandre des centaines de milliers de "fake news" à une fréquence de plus de trois millions de postes par minute. Deuxièmement, les responsables de la diffusion de ce contenu frauduleux exploitent la puissance de modèles d'intelligence artificielle pour conduire leurs attaques et déguiser une fausse information en un article standard. Troisièmement, des utilisateurs innocents, s'ils ne sont pas alertés par un modèle algorithmique, peuvent accidentellement répandre de fausses informations dans une vague exponentielle de partages, de postes et d'articles. La vague de mésinformation est souvent détectée lorsqu'elle est déjà hors de contrôle et peut, par conséquent, causer de graves dommages dans un court lapse de temps.

Le but fondamental de cette thèse est d'aider les organisations médiatiques, les gouvernements, le public et l'académique à encadrer le développement de la lutte contre la mésinformation, en intégrant entre les modèles algorithmiques et les validateurs humains, tout en contrôlant les coûts, afin de prévenir des dommages liés à la mésinformation. Cette thèse abord trois composants essentiels de cet encadrement : (i) détection, (ii) validation et (iii) visualisation. Pour chacun d'eux, nous nous concentrons sur un domaine de la mésinformation, ce qui nous permet l'adoption d'une approche

systématique dans le développement de méthodes dédiées afin de résoudre le problème sous différents angles.

Les contributions principales de cette thèse sont:

- **Détection:** La détection précoce peut potentiellement prévenir la propagation de la mésinformation en identifiant les nouvelles suspicieuses et en les portant à l'attention de validateurs humains ; dans ce but, nous proposons un modèle progressif, basé sur les graphs, qui détecte l'émergence d'articles de désinformation à partir de flux d'actualité ou de flux de médias sociaux. Le modèle a confirmé et exploité l'effet de "chambre d'écho" propre aux vagues de mésinformation pour identifier les entités affectées sur les réseaux sociaux à travers leurs interactions.
- **Validation:** Apprendre un bon modèle de détection demande un grand volume de données; cependant, il peut être mis à jour rapidement en intégrant de nouvelles tendances sociales. Une approche prometteuse fait usage d'experts humains pour valider les résultats de détection, aidant ainsi les modèles algorithmiques à devenir plus performants et à s'adapter aux nouvelles caractéristiques de la mésinformation. En adoptant cette approche, nous proposant une stratégie de guidage qui minimise l'effort de validation des alertes de mésinformation, améliorant ainsi la confiance dans la détection de contenu frauduleux et réduisant les risques de fausses alertes.
- **Visualisation:** Disséminer les rapports concernant les articles frauduleux est une étape essentielle pour rendre le public attentif à la mésinformation et éduquer les utilisateurs du Web. Cependant, les utilisateurs peuvent être rapidement dépassé par le volume d'information titanesque provenant du Web, surtout lorsque le niveau de redondance augmente et la densité de la valeur diminue. Dans ce contexte, nous proposons un protocole de rétention de flux de données afin de visualiser de l'information représentative avec un regret minimal. La mesure d'utilité est établie de façon à ce que les données couvrent les sujets de discussion émergents, les postes récents provenant des réseaux sociaux et possédant un contexte social riche.

En résumé, cette thèse propose les composants clés visant à construire un cadre pour la lutte contre la mésinformation. Les techniques proposées améliorent les résultats des méthodes existantes dans divers domaines, incluant les rumeurs, les affirmations sur le Web et les flux de données provenant des réseaux sociaux.

Mots-clés: *mésinformation digitale, détection d'anomalie, minimisation de l'effort, analyse de médias sociaux, visualisation de flux de données*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The use of freely available online data is rapidly increasing, as companies have detected the financial value and potentials of these data in their businesses. In particular, data from social media have attracted an enormous amount of interests in the recent decade, as they can, when properly treated, assist in achieving customer insight into business decision making. However, the distributed and decentralized nature of this kind of user-generated content presents a new kind of challenge: information is largely propagated without any filters for quality control. This leads to a large variety of misinformation on the Web, particularly on the social media landscape, such as false news, satire news, and rumours [Zea18].

1.1 Motivation

Our modern society is struggling with an unprecedented amount of online misinformation, which do harm to democracy, economics, and national security [VRA18]. Creators of misinformation optimise their chance to manipulate public opinion and maximise their financial and political gains through sophisticated pollution of our information diffusion channels. The Digital News 2018 Australian report shows that three quarters of online news consumers say they encounter one or more instances of fake news every day [oC]. Journalism and politics have been impacted by fake news on a global scale, with weakened public trust in governments seen during the Brexit referendum and viral fake election stories outperforming genuine news on social media during the U.S. presidential election campaign [New]. Online misinformation also single-handedly caused $136.5 billion in losses in the stock market value through a single tweet about explosions in the White House [ZAB+18]. Such attacks are even driven by the advances of modern artificial intelligence (AI) these days [cfr] and pose a new and ever-evolving cyber threat operating at the information level, which is far more advanced than traditional cybersecurity attacks at the hardware and software levels [gov].

Research in this area is still in its infancy but demonstrates that preventing the spread of misinformation is a formidable challenge. This is due to several reasons. First, the open nature of social platforms such as Facebook and Twitter allows users to freely

produce and propagate any content without authentication, and this has been exploited to spread hundreds of thousands of fake news at a rate of more than three million social posts per minute [All17]. Second, those responsible for the spread of fake news harvest the power of AI attacking models to mix and disguise falsehoods with common news. Methods of camouflage are used to cover digital footprints through synthesising millions of fake accounts and appearing to participate in normal social interactions with other users [HSB+16]. Third, innocent users, without proper alerts from algorithmic models, can accidentally spread misinformation stories in an exponential wave of shares, posts and articles. The misinformation wave is often only detected when already beyond control and consequently can cause large-scale effects in a very short time. Early detection can potentially prevent damage from occurring by flagging suspicious news for human attention; however it remains, to date, an unsolved challenge.

Existing techniques for detecting misinformation in online social networks are focused on building fully autonomous algorithmic models [Zea18]. However, such models become obsolete with the new generation of AI-driven attacks. Fabricated videos using AI to mimic real people such as Barack Obama as well as social media bots and clickbaits disguised to appear real can go by unnoticed with existing techniques [cfr]. Detecting this new generation of misinformation requires a deep understanding of social contexts, which often exceeds the limits of autonomous algorithms. Learning a near-exact algorithmic model, even AI, already requires a huge amount of training data; yet this data is often not available in advance and can be outdated swiftly with new social trends [LNL+15]. Furthermore, giving algorithmic models the privilege to be a judge of truth in modern society has raised ethical concerns regarding fairness and transparency [HBC16], i.e. who is there to check them? A promising approach is to use human experts to validate the algorithmic detections, boosting the confidence of misinformation alarms. In this research, we bring humans on board by harnessing the advances of human validation platforms such as Snopes and Figure-Eight, which provide tools and incentives to employ millions of experts for any validation tasks.

1.2 Misinformation Debunking Framework and Research Questions

The overarching goal of this thesis is to build the cornerstones of a human-powered misinformation debunking framework for helping media fight misinformation and restore public's trust with cost-effective, scalable, robust and streaming techniques on top of large-scale dynamic social networks. The debunking framework is expected to work on data streams of social platforms and issues alarms on suspicious information waves. Human experts are employed to validate these emerging stories, guiding algorithmic models to train themselves to become smarter and adaptive to new traits of misinformation formation. We argue that a typical debunking framework would require three important components: *Detection*, *Validation*, and *Visualisation*, as illustrated in Figure 1.1.

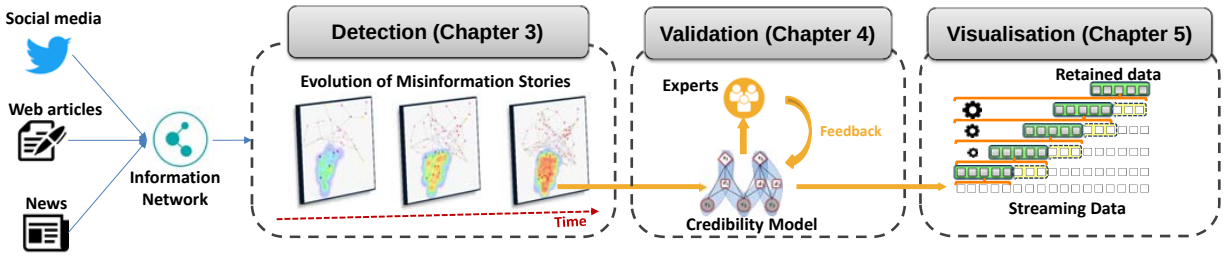


Figure 1.1: Misinformation Debunking Process

Detection. The role of this component is to develop a model that can detect emergent misinformation stories from data streams of social networks. In particular, the early detection will help preventing the spread of misinformation from occurring by flagging suspicious news for human attention. To achieve this goal, there are numerous research questions to tackle:

- *How to identify the indicative signals for distinguishing misinformation from genuine news?*
- *How to detect the affected social entities with high confidence?*
- *How to design a progressive detection algorithm that works with data streams of social platforms?*

Validation. The role of this component is to build a seamless and cost-effective integration of human validators to power the algorithmic models in improving detection accuracy and reducing the risk of false alarms. To achieve this goal, the following research questions need to be answered:

- *How to select the most beneficial question(s) for human feedback?*
- *How to design support information to increase feedback quality, avoiding mistakes and biases?*
- *How to incorporate user input to improve the correctness of credibility model?*
- *How to define termination criteria to get the best trade-off between validation time and detection accuracy?*

Visualisation. The role of this component is to design a data stream visualisation protocol that addresses the issues of high velocity and high volume with massive social streams. Representative social entities should be retained while the lesser ones should be discarded to preserve the framework storage. To achieve this goal, the following research questions are required to solve:

- *How to design a utility function that captures different aspects of social data?*

- *How to build an information sketch to help computing utility accurately even when historical data is discarded?*
- *How to develop a retaining algorithm that preserves a part of historical data with minimal information loss?*

1.3 Thesis Methodology

The theme of this thesis is to treat the misinformation debunking framework from different angles; so that when working with different types of misinformation across different social platforms, end-users can reuse the developed models to discover domain-specific insights and practical guidelines. To do so, we follow a bottom-up approach, where each of the framework components (*Detection*, *Validation*, and *Visualisation*) focuses on a well defined working domain (rumours, online facts, and social streams), for which the required data is accessible, and that is representative of misinformation landscape, allowing us to explore various types of challenges arise. While there are limits to such an approach, as the working domain might under- or over-represent their kind, by allowing researchers to focus on a common set of problems and data, it facilitates a better understanding of fundamental mechanisms and their guarantees.

1.4 Contributions and Thesis Organisation

In addressing the above research questions, this thesis makes the following contributions:

Misinformation Detection: The Case of Rumour Early-Detection. In Chapter 3, we solve the problem of *Early Detection on Dynamic Data Networks*. That is, finding a set of entities on a social graph that are affected by a rumour propagation. In particular:

- We develop a model that grounds rumour detection on a generic graph representation of social data, thereby achieving a solution that is applicable for any type of social platform.
- Based on a model for social platforms, we develop a general process to detect rumours by observing anomalous signals indicative of rumours.
- We show how to apply our approach for streaming data by incrementally computing anomaly scores on the local level and global level.

Misinformation Validation: The Case of Minimal-Effort Fact-Checking. Expert input is expensive (in terms of time and cost), so that a validation of all misinformation claims is infeasible, even if one relies on a large number of experts. There is a trade-off between the precision of a credibility model and the amount of expert input: the more claims are checked manually, the higher the precision. However, expert input is commonly limited by some budget. Chapter 4 solves the problem of *Human-powered Validation with Minimal Effort*.

- We design a cost-effective iterative process for guiding users in fact checking.
- We present a novel probabilistic model that enables us to reason on the credibility of claims, while new user input is continuously incorporated.
- We propose strategies to guide users, i.e., to select the claims for which validation is most beneficial. These strategies target the reduction of uncertainty in our probabilistic model for fact checking.
- We combine our mechanisms for credibility inference and user guidance to obtain a comprehensive validation process. We also show how to achieve robustness against erroneous user input.

Misinformation Visualisation: The Case of Minimal-Regret Data Streaming Retaining. In Chapter 5, we solve the problem of *Progressive Data Visualisation with Minimal Regret*. We consider the natural setting of social platforms, where data is dynamic and available as a stream. Then, retaining of data becomes more challenging compared to one-off summarization, as data selection has to be repeated every time new data arrives. Instead of considering the whole historical data, summarization now works on the retained data (i.e., a previous summary) and the new data.

- We propose a novel statistical model, which does not only capture the traditional context of social data (importance of topics, user influence, information diffusion), but also embeds the dynamics of this context over time.
- We design a utility function to assess the representativeness of a subset of data items against all historical data.
- We develop a progressive algorithm to solve the streaming data visualisation problem such that the retained data has minimal utility loss.

The remainder of this thesis is organised as follows. Chapter 2 presents a survey of literature related to research challenges addressed in this thesis work. Chapter 6 concludes our thesis and discusses the future work.

1.5 Selected Publications

This thesis is based on the following research papers:

- Nguyen, T.T., Weidlich, M., Zheng, B., Yin, H., Nguyen, Q.V.H., and Stantic, B., 2019. *From Anomaly Detection to Rumour Detection using Data Streams of Social Platforms*. In the 45$^{\text{th}}$ International Conference on Very Large Data Bases. (**VLDB 2019**)

This paper presents models and methods to realise the idea of detecting rumours based on anomalies. It follows a data management approach: rumour detection is grounded in algorithms that work on a generic graph representation of social data, thereby achieving a solution that is applicable for any type of social platform.

- Nguyen, T.T., Weidlich, M., Yin, H., Zheng, B., Nguyen, Q.V.H., and Stantic, B., 2019. *User Guidance for Efficient Fact Checking.* In the 45th International Conference on Very Large Data Bases. (**VLDB 2019**)

  This paper proposes a comprehensive framework to guide users in the validation of facts, striving for a minimisation of the invested effort. The framework is grounded in a novel probabilistic model that combines user input with automated credibility inference.

- Nguyen, T.T., Phan, T.C., Nguyen, Q.V.H., Aberer, K. and Stantic, B., 2019. *Maximal Fusion of Facts on the Web with Credibility Guarantee.* Information Fusion Journal, 48, pp.55-66. (**IFJ 2019**)

  This paper overcomes the inherent trade-off between the precision of information credibility and the recall of information coverage in a novel way for sensitive applications: maximizing the recall while preserving the precision at least better or equal to a pre-defined requirement.

- Nguyen, T.T., Duong, C.T., Weidlich, M., Yin, H. and Nguyen, Q.V.H., 2017. *Retaining data from streams of social platforms with minimal regret.* In the 26th International Joint Conference on Artificial Intelligence. (**IJCAI 2017**)

  The data streams of today's social platforms exceed any reasonable limit for permanent storage, especially since data is often redundant, overlapping, sparse, and generally of low value. This paper proposes techniques to effectively decide which data to retain, such that the induced loss of information, the regret of neglecting certain data, is minimized.

Chapter 2

Background

In this chapter, we review the literature related to this thesis work. For a better understanding with clear organization, we present the following topics as the research background of this thesis; i.e.,

2.1 Misinformation Landscape

2.1.1 Social Data and Social Platforms

Online social data typically includes digital traces generated by (or about) Web users, providing insights into how people behave, communicate, and interact in real-world [KSG13]. It has been coined a variety of terms such as "crowdsourced data", "wisdom of crowds", "human traces", "usage data" by the community to describe its collective and user-driven nature [BY14, Olt16]. One origin of social data is from people who voluntarily produce content by reporting scientific studies, uploading their comments, writing product reviews, and sharing knowledge via various Web platforms, such as blogs (e.g. Tumblr, Wordpress), social media (e.g. Twitter, Facebook), and wikis (e.g. Wikipedia, Wikirate) [NPN+19, HWN+19, NNL+19, NYW+19, NWZ+19]. Examples of social data and their statistics can be found in Table 2.1.

Table 2.1: Examples of online social data

Data sources	Size	#Users	Content
Twitter [Twi]	$\sim$ 0.5B tweets/day	$\sim$ 0.3B active users	opinions
Tumblr [Tum]	$\sim$ 100B posts	$\sim$ 0.25B blogs	arguments
Wikipedia [Wik]	$>$ 35M articles	$>$ 70K active contributors	facts

The attention around online social data has particularly grown with of a diversity of social platforms, from social media sites (Twitter) to social networks (e.g. Facebook), from consuming information to interacting with friends, resulting in the proliferation of social units (e.g., clicks, likes, shares, social links) [Tuf14]. While social data itself is not the origin of misinformation, malicious users can easily leverage those social units to spread and amplify misinformation, as well as other forms of falsehoods such as fake

reviews, spams and scams, and propaganda [VRA18]. This thesis shows that those social units are not as bad as everyone thought and can be used together to trace and detect emergent misinformation stories.

2.1.2 A Taxonomy of Misinformation

There are a plethora of definitions have been attempted to categorise misinformation. A popular one is information authenticity approach, where the veracity is emphasised rather than the intentions of misinformation creators. In this approach, online misinformation has been connected to various terms and concepts such as maliciously false news, false news, satire news, and rumour news [Zea18]. Here, the term 'news' broadly refers to all kinds of claims, statements, speeches, posts via social media or mainstream channels. This definition supports most existing fake-news-related studies, and datasets, as provided by the existing fact-checking (aka misinformation debunking) websites (e.g. Snopes) [ZZS+19].

Another categorisation is information intention approach, in which misinformation is defined as incorrect information with no intention of harm. Whereas, disinformation is incorrect information with the intention of deceiving the consumer [CGL+18]. In either cases, the incorrect information can be accidentally shared by social users, blurring the line between innocence and deception. In this thesis, we refer the term 'misinformation' as an information pollution phenomenon that disrupts our information diffusion channels, including propaganda, rumors, and misleading reports. Powered by our ubiquitous Web technologies and modern artificial intelligence, these types of falsehoods can be fabricated easily and realistically, and be propagated with unprecedented speed and scale [ZZJ+19, TPN+18, HVN+18, DLV+18, TPT+18]. The damages are becoming more catastrophic, including sensitive domains such as vaccination and political elections, affecting many lives [ABC+18].

2.2 Web Credibility

Web credibility and misinformation detection are two sides of the same problem on curating information on the Web. While Web credibility research generally focuses on building a reputation system where each source and each data item is assigned a credibility score indicating the correctness of its information capability, misinformation detection approaches the problem from a data cleaning perspective, where errors, noises, and biases in social networks are identified and removed promptly to avoid 'domino' effects. In this section, we review the literature of web credibility on how the Web data is wrangled, evaluated, and aggregated to build credible Web knowledge bases.

2.2.1 Information Extraction

Fact extraction. Fact extraction may be performed by diverse data representations, e.g., knowledge bases [DGH+14], web tables [CHW+08], semi-structured data [ECD+04],

or free text [BCS+07]. Other work uses co-occurrence information and evidential logs [LDLL17, LMY11], but is limited to quantitative information such as identifying unpopular facts based on the number of mentions [LDLL17, DCMT19]. Various tools such as TextRunner [YCB+07], TweetIE [BDF+13], and DeepDive [DSRR+16] have been developed to organise non-structured facts (e.g. textual statements) into structured data, particular relational tables that capture the relationships between information units [SS17].

Our work is orthogonal to all the above mentioned. By relying on an abstract data representation, our model is not specific to a particular domain. Our principles of user guidance can further be adapted for many of the above techniques, exploiting its generic notion of uncertainty.

Credibility indicators. A lot of prior research have been conducted on identifying credible claims from the Web [PROA12, YJP17]. In these works, they capture the credibility of a claim as a combination of individual features/indicators. The first type of features is content-based, such as semantic features (e.g. category, entities, keywords), sentiments features (e.g. subjectivity), and syntactic features (part-of-speech tag, punctuation marks, spelling errors), advertisements, and page layout [VLCH18, CCWH18, PCBH17, Cam16]. The second type of features is network-based, such as the overall ratings of sources sharing the same claims. However, most of the existing works only compute the credibility as an aggregation function of these features; and again, do not provide a precision guarantee. On top of these works, we reuse these features and additionally take into account the mutual relationships between sources, their documents, and claims by the factor graph model to support the precision guarantee process.

2.2.2 Data Representation

Information networks. There exists various graph-based models for data of social platforms, referred to as information networks [SLZ+17]. Some models capture real-world entities, such as users and posts [SH12], while others represent derived data elements, such as topics [TQW+15]. Data representation for special types of misinformation such as fake news, in which the propagation of fake news in social networks is modelled by *information cascade* [FAEC14], a tree-like data structure rooted from the genesis post. Existing work on misinformation detection in information networks focuses on modelling the propagation patterns of known phenomena [FAEC14, ZAB+18] or classifies known events [ZRM15]. This setting is orthogonal to our work, since we strive for the detection of phenomena that emerge on social networks, but are not known a priori [PZH+18, NZW+18a, YZN+18, NVN+18, PSH+18].

Belief databases. Various efforts have been spent to build databases of knowledge including facts (objective), claims (unverified), and beliefs (subjective) [DGM19]. In particular, healthcare-related facts (e.g. side effects of drugs) have been extracted from online forums and validated by an expert clinic portal [MWDNM14a]. Subjective facts are also studied, in which arguments (claims + evidences) supporting a user point of view are collected for modelling the opinion landscape in online social data [STVB16]. A

special type of belief in relational model is also studied, which indicates the confidence of an attribute value of database entities (e.g. user profile, background information) [GS10]. In that, a relational model has been built to capture the relationship between users and their believes, reflecting how the belief is transferred from one user to another [GBKS09].

2.2.3 Trust Computation

Credibility Assessment. The study of trust computation on the Web in general and social networks in particular has been concerned with how to evaluate the credibility of data items and the trustworthiness of data providers [Wie18, TVF+17, HVT+18]. Existing techniques for this include: (i) *statistical approaches* – Bayesian systems and belief models that compute continuous trust values [HVT+18], (ii) *machine learning techniques* – artificial neural networks and hidden Markov models that compute discrete trust labels [QLNY18], (iii) *heuristics-based techniques* – practical consistency rules for real-time systems [HDT+17], and (iv) *behavior-based models* – user pattern (e.g. activity frequency) [YLLL17]. These techniques are, however, static and require a large collection of data in advance. They cannot produce timely alarms in response to the arrival of a stream of data [PTHS18, HDT+17, HTN+17, DNWS17, YCS+17].

Techniques for credibility assessment vary from domain to domain, including medical articles [NBW18], Web pages [KNW17], social media [SCA19], and subjective ground-truth (user opinion and sentiment) [THL+15]. For example, [SCA19] uses published scientific articles to verify the credibility of social media news by computing textual similarity. Following a similar but more fine-grained approach, [NBW18] splits the documents into sentences and comparing these sentences with a knowledge base of experts' medical statement. On the other hand, [KNW17] leverages the 'wisdom of the crowd' via crowdsourcing (Amazon Mechanical Turk) to provide labels on dozens of criteria (freshness, informativity, references, language quality, contact information, etc.) and then aggregate all of these factors into a credibility rating on a five-point Likert scale for a Web site [TWT+17, YZC+16, PH16, YHZ+16, HTNA15].

Truth finding on the Web. Given a set of claims of multiple sources, the truth finding (aka fact checking) problem is to determine the truth values of each claim [DSS12]. Existing work in this space also considers mutual reinforcing relations between sources and claims, e.g., by Bayesian models [ZRGH12], maximum likelihood estimation [WKLA12], and latent credibility analysis [PR13]. However, these techniques neglect posterior knowledge on user input and rely on domain-specific information about sources and data, such as the dependencies between sources and temporal data evolution [DSS12].

Truth finding is also known as *knowledge verification* [LDLL17] and *credibility analysis* [MW15]. Existing automatic techniques mostly look at features of data, such as number of relevant articles, keywords, and popularity, which are noisy and can be easily dominated by information cascades [LDLL17]. Again, posterior knowledge on user input cannot be incorporated. Also, approaches based on gradient-descent [MW15]

only optimise model parameters, but neglect external probability constraints [TNHA15, NHWA15, HNC+15, HTWA15b, HTWA15b].

Truth finding is also studied in relational databases [CMT18]. For enterprises, this includes finding the emergent semantics across department and sector [ACMH03, ACMO+04]. However, the schemas of these database sources are heterogeneous, due to the fact that they are designed in different time, by different people, and for different purposes. This requires further data preprocessing steps such as data integration [HWT+19, NNM+14, BH08] and data cleaning [YEN+11]. For Deep Web (i.e. Web tables Web [NNWA15] that are generated by underlying relational databases), truth finding is hindered by an additional processing challenges including unstructured format, data extraction error, and restricted access to original data sources [LDL+13]. For Argument Web [RZR07], truth finding includes the resolution of controversial issues, in which finding and aggregating evidences for unverified claims is the first citizen. However, while some claims can be objective verified (e.g. climate change), a large amount of them are subjective [BLSR13].

Fact Checking. The fact checking literature, in particular the claim accuracy assessment, focuses on the classification of claims by credibility, based on a fixed training data [CLL+18b, CLL+18a, LMT18, Man19]. This can be seen as the starting point for our work: We put an expert user in the loop to clean the results obtained by automated classification. Our guidance strategies therefore complement the literature on classifying claims in identifying which potential errors of a classifier are most beneficial to validate by an expert user. At the same time, our approach can also support an expert user in building up a fact database from scratch, in a pay-as-you-go manner. Moreover, our approach goes beyond recent work on offline fact checking, e.g., [PMSW17], by including a streaming process to incorporate new claims on-the-fly [HTWA15a, HNMA14, HSDA14, HTNA14, HNM+14].

Fact checking is also studied from many different perspectives, such as content management [Man17], query processing [BCC+16] (e.g. to check relationships between claims in their RDF representation), and scalability [Man19]. In particular, scalability is a key issue since it is hard to construct a database of verified social facts reaching the scale of millions, due to various content management issues such as data management, data integration, NLP, text analysis, and graph mining [CLL+18b]. This hinders the power of algorithmic models in learning both generic and domain-specific rules for assessing the credibility of claims. To facilitate the scalability problem, some pioneering efforts have been spent to develop crowd-sourced platforms that use the mass of crowd inputs to build large-scale fact data [Fis], but still in early development [HJA13, HLM+13b, HNMA13, HWM+13, HLM+13a, GKS+13].

2.3 Misinformation Detection

2.3.1 Intradisciplinary Approaches

Classification as Detection. Existing works on misinformation detection on social platforms focus on classification of social news as false or not, using techniques grounded in detective features *manually* defined from statistical observations and domain knowledge [ZAB+18, SSW+17]. To avoid ad hoc definition of these features, deep learning methods have been applied to learn them *automatically* by extracting temporal dependencies and the dynamics of misinformation propagation [Zea18, MGW+15]. Real-time detection models were studied, but not adaptive to new traits of misinformation that appear outside their knowledge base [LNL+15]. In general, existing techniques require a preprocessing step to explicitly group social posts into a news event, which may have a cumulative effect on inaccuracies in the main detection step as well as posing an inconvenience for real-time data stream processing. Moreover, the computed features are often not embedded in the same value space, and thus not generic enough to fuse several kinds of misinformation indicators [HNLA13b, HA13, HNLA13a, NHQ12, HA08].

Rumour detection. Rumour is a domain-specific term of misinformation, in which false information often originates from the fact that people tend to exaggerate what they dislike [TWZ+19]. While there is a large body of work on rumour detection on social platforms, surveyed in [ZAB+17], little has been done to exploit multiple modalities to detect rumours. Most work leverages only textual data such as tweets [CMP11, ZRM15, GKCM14]; whereas others consider different data entities such as users and hashtags but still treat them as additional features or textual data only [MGM+16]. Techniques based on hand-crafted features [CMP11, ZRM15, YLYY12] are grounded in an ad-hoc definition of features, which are expected to be strong indicators of rumours. Recently, deep features based on temporal dependencies of the posts have been proposed [MGM+16]. While this approach achieves high detection accuracy, it first requires the detection of an explicit event and thus depends on the accuracy of this event detection step. There are further approaches [MGW17, WYZ15, WT15] that mine behavioural patterns of social users by extracting subgraphs of social posts (support, deny, question) on how rumours propagate [ATH06]. However, these techniques require large collections of tweets to conduct the respective analysis. As such, they cannot be expected to yield small lag times in the detection of rumours and are not well-suited for a streaming setting. Our approach is the first to leverage not only the textual data, but also other modalities in both offline and online settings.

Anomaly detection. Anomaly detection can be classified into point or group-based techniques [YQW+16]. Point-based anomaly detection aims to detect individuals, for which the behaviour is different from the general population [SDJ+01, JV01, IHS06]. Group-based anomaly detection, in turn, strives for groups of individuals that collectively behave differently compared to some population [CBK07, DSN09, CN14, YHL15, MS13, XPS+11]. However, none of the above techniques has been applied to rumour detection.

While [CN14] addresses a similar use case, it neglects the anomalies related to feature differences between entities. Our technique is the first one for group-based anomaly detection that simultaneously identify anomalies in all features, entities, and relations. Most of the work on anomaly detection in general and rumour detection in particular focuses on accuracy. Here, we define the detection coefficient to capture the balance between accuracy and completeness, which is optimised by our approach.

2.3.2 Interdisciplinary Approaches

Misinformation Detection in Journalism. Computational journalism is an established area to support journalism work with computational methods [CHT11, CLYY11]. In that, hard or verified facts are the first-class citizens; and the task of finding them becomes automated for scale. In mainstream media, such task is guided by domain experts, where reputation and accountability are frequently validated, cross-checked and guaranteed. As such, misinformation detection in journalism rather is rather about detecting mistakes and processing errors during the publication processes [HSW+14]. In particular, computational journalism research focuses on extracting and exploring statements (e.g. knowledge triples) from the Web to support journalists in forming evidences and claims. The veracity of Web statements are not heavily focuses and often implicitly verified by domain experts (i.e. the fact that they do not use some statements for their arguments might indicate that they are not true).

With the advent of Web technologies, in particular social media, every internet-user can become his own journalist, threatening the guarantees of long-established credibility and accountability of traditional journalism. The need of assessing data sources and computing the credibility of its claims (commonly called as "facts") becomes the heart of misinformation detection, web credibility, and fact-checking research [SS17].

Misinformation Detection in Psychology. While creators of misinformation might only produce the genesis post, online misinformation is, in fact, spread by social users. Understanding the psychological factors would help to understand and predict the propagation behaviour of misinformation stories. Research in social psychology proved that in average, humans can only detect falsehoods better than a coin-flip by 5%-8% [Rub10]. This result has a profound implication, especially with misinformation stories intentionally fabricated for deception such as fake news, that the social users can be easily manipulated. Even worse, the success of deception can be increased further by validity effect [Boe94] (humans can easily trust a story if being exposed to it several time), by confirmation bias [Met13] (humans can be easily deceived if the misinformation is built on top of their prior knowledge), or by bandwagon effect [Lei50] (you tend to believe if most of your peers already believe) [Zea18].

Misinformation Detection in Economics. Modeling the propagation of misinformation stories in social networks is an important task in detecting and predicting

their social damages. This is done by various quantitative models such as social influence theory [DLBS14], information diffusion framewokr [NDG12] and forecasting methods [DS14]. In addition, economic-related model such as epidemics is proven to help increase the performance of quantitative tools [Zea18], since the propagation of information in human society, especialy misinformation stories, is similar to how contagious diseases are spread. For example, information diffusion model in social networks can be augmented by a heat kernel function to control the propagation rate. Economic-related game theories can also be applied to model the dominance of an information source over another via information utility, psychological utility, short-term economical utility (e.g. profit), and long-term social utility (e.g. reputation) [Zea18].

2.4 Human-powered Validation

2.4.1 Human Validation Platforms

Human validation platforms have drawn attentions in both academia and industry due to their ability to employ millions of online workers in real-time [TTFS18]. Crowdsourcing marketplace such as Amazon Mechanical Turk is one type of such platforms, in which a freelancer flags malicious contents such as frauds and scams to help machines learn an accurate detection model [HVT+18]. Fact checking site such as Snopes and PolitiFact is another type, where journalism experts are employed to verify rumours [TWY+19]. Despite of such availability, existing human validation solutions cannot be tailored easily for misinformation debunking due to the domain-specific challenges of task design, task scheduling, task distribution, and consensus computation [HVT+18]. This thesis brings novelty to the field by minimising the human efforts needed to validate misinformation flags, which are detected by the *Detection* part of the thesis.

Micro-work systems. In micro-work systems, a crowd of users is employed to complete small tasks (ranging in time duration from a few seconds to a few minutes) for fair amounts of incentives. These systems provide an opportunity for time and money work that would otherwise be accomplished by a handful of hired experts, to be completed in a fraction of the time and money by a crowd of ordinary humans. For example, Amazon Mechanical Turk (AMT) is a well-known micro-work system with millions of active crowd workers and hundreds of thousands of human computation tasks. In Amazon Mechanical Turk (AMT), tasks range from labeling images with keywords to judging the relevance of search results and linguistic jobs (e.g. translate, proof-reading). Although workers from any country can work on tasks on AMT, only US and Indian workers can receive money directly in their bank accounts [RIS+10]. Therefore, the majority of workers on AMT are US and Indian citizens. Moreover, the workers are young since more than 50% of AMT workers have age below 34 [RIS+10]. In addition, the workers keep getting younger as the average age decreases through time. The workers on AMT are highly educated since most of them have an undergraduate degree or higher [RIS+10]. Other well-known micro-work platforms include CrowdFlower, CloudCrowd, etc.

Implicit human computation also involves the completion of micro-tasks by crowds of human users. In implicit human computation users solve a problem as a side effect (passively) of something else they are doing. The ESP Game [VAD04] provides an example of an implicit HC system that allows people to label images while enjoying themselves. In this game, the participant labels an input image by a keyword, which most properly describes the image, from a set of provided keywords. The ultimate goal is to obtain proper labels for each image. This effort is part of the larger goal of collecting proper labels for images on the Web, which would be an invaluable for information retrieval applications.

Another example is reCAPTCHA [VAMM+08], which is a piggyback HC system built on top of CAPTCHA (used by websites to prevent spam). By solving the CAPTCHA, users implicitly perform OCR tasks, such as digitizing books, newspapers and old time radio. In a reCAPTCHA task, a user is presented with two words. One word serves as a conventional CAPTCHA, while the other word cannot by recognized by automatic OCR techniques. If a user recognizes the recognized word, the answer to the unrecognized word is assumed to be correct, and is collected as training data for further OCR tools.

Social-based systems. Social-based human computation systems encourage millions of people over the world to contribute to human computation problems via the Internet. With the growth of Web 2.0 technologies, there are many kinds of works that can be performed by people. The first type of social-based HC systems can be described as 'Knowledge-Base'. Wikipedia is a well-known example of human computation knowledge-base, which has thousands of editors to continually edit articles and contribute knowledge for building the world's most comprehensive free encyclopedia. The writing is distributed and open in that essentially almost anyone who has access to the Internet can contribute.

Q&A sites that allow users to post questions, provides answer, and edit and organize the constructed information, also fall under the category of Social-based HC systems. A typical example is Yahoo! Answers, which is a general Q&A forum. The human computation in such systems involves answering the questions and the incentives include social benefits such as prestige, fun and social networking. The collected human answers provide a large body of human-knowledge data, that can be used for solving further AI problems. Other example Q&A sites are ask.fm, Quora, etc.

Some social-based HC systems take the form of competitions. In contrast to micro-tasks, HC competitions aim to solve complex problems and offer high prices. Instead of collaborating, human participants compete with each other to achieve higher ranks. The collected human solutions offer a great variety of ways to solve a particular computational problem, thus not only enhancing overall knowledge but also changing the way computers approach the problem. A well-known example is Topcoder.com, which offers various computer science problems, ranging from algorithms to software design and development. Another example is the Goldcorp Challenge [1], which employs geological experts from all over the world to identify the locations of gold deposits.

[1] http://www.goldcorpchallenge.com/

'Crowdfunding' and 'Skill Markets' are yet other types of social-based HC systems. Crowdfunding is an HC-based strategy for funding one's projects by asking a multitude of people to contribute small amounts of money instead of seeking huge contributions from a few big investors. The advantage of crowdfunding is that it is easy for people to invest a small amount of money. A multi-level payment mechanism is often applied, in which the more one contributes, the more rewards one gets such as e.g., souvenirs, progress updates, first copy of the product at discounted price, etc. In 'Skill Markets', people work as freelancers to complete jobs. Example marketplaces include Elance, ODesk, and Freelancer.com, where millions of freelancers do various kinds of work.

It is worth noting that social-network services are also considered as social-based human computation systems due to their long-existing nature of human computation. They enable the wisdom of the crowd efficiently by providing fundamental infrastructure to employ a mass amount of user in a short period of time. As such, social-network services are born a platform qualified not only for spreading human communications, but also for human computation tasks. For example in [CSTC12], the authors leverage a micro-blog service (i.e. Twitter) to collect answers for decision making questions by actively distributing the questions to workers via the "@" markup. However, social network services are only best-fit for knowledge collection. This is because it is often difficult to build a payment mechanism on top of social networks. As such, social networks cannot be used for micro-tasks that require monetary payment.

Pervasive systems. Pervasive systems make use of activities that human beings perform in their daily lives to solve computational problems. With the rapid uptake of mobile technologies, we can access human computation power via smart phones to implement pervasive systems. The first type of such systems include community-based traffic navigation platforms (e.g. Waze), also called geosocial networks. In Waze [FKP+12], the human participants are drivers who report real-time traffic and road information such as accidents, traffic jams, speed-traps, and nearby police units. All this information is publicly shared among drivers for the purposes of routing and navigation. The Waze community has millions of active users mainly across Europe, Asia, and North America. Similarly, Google Maps and Google Earth also employ human-powered traffic information for various data visualization purposes.

Human-powered newspapers serve as another example of pervasive HC systems. Instead of using professional reporters, local people are employed to report rumours and stories in their communities. An example is Ushahidi [Oko09], in which people in crisis situations (e.g., in disaster and conflict zones) submit their reports through the web and mobile phones. These reports are then aggregated and organized temporally and geospatially to give a general view of emerging situations. There are some distinct advantages of this approach. One is the relatively faster reporting of information as compared to traditional methods since professional journalists are limited in number. Also, in human powered newspapers, the information is untamed and covers different points of views of human participants, as opposed to the point of view of a single individual (the professional reporter).

Participatory sensing systems are also pervasive HC systems, in which people equipped with sensors, e.g., built in their smartphones, measure environmental conditions. An example is Common Sense [DAK+09], which is a human-power pollution monitoring application. Common Sense uses specialized handheld air quality sensing devices, which are deployed across a large number of human participants, to collectively measure the air quality of an area. Similarly, we can apply the same method to monitor other environmental conditions such as noise and water.

Management systems. Management systems are general-purpose human computation systems that are designed to manage the entire human computation process, including designing and posting tasks, collecting and aggregating human inputs, and performing further analyses. The first example is CrowdDB [FKK+11], which aims to develop a declarative language to express the logic of the expected human computation task instead of describing it in natural language. CrowdDB employs human power via crowdsourcing to answer the uncertain queries that cannot by processed by automatic engines.

Another example is CrowdForge [KSKK11], which aims to manage human computation workflow, including decomposing large tasks into small ones, and assigning these tasks to human participants. Both dynamic and fixed workflows are supported to allow the parallelization of human computation tasks. More precisely, CrowdForge employs a MapReduce-liked model to post micro-tasks into crowdsourcing platforms such as Amazon Mechanical Turk. A sample complex task studied is article writing [KSKK11], in which an article is partitioned into different Map tasks, such as collecting and writing facts about an entity. After all facts are collected, a Reduce task is performed by human users to combine the facts into one paragraph. For quality control purposes, CrowdForge uses majority voting to determine the best write-ups for the article.

As an industrial example, CrowdFlower [DWKDH15] (now becomes Figure-Eight [Eig18]) supports integrating human computation into business processes by offering three important features: workflows, taxonomy, and quality control. First, CrowdFlower help users define the workflows by pre-designing job templates for crowdsourcing tasks such as image categorization, text transcription, and sentiment analysis. On top of the user-defined workflows, the system will automatically route and process data between multiple CrowdFlower jobs and/or external services. Second, CrowdFlower help users manage a large number of jobs hierarchically by letting them define a taxonomy of tags and indexing the CrowdFlower jobs by these tags. Based on the tag index, users can search the jobs efficiently. Third, CrowdFlower allows to control the quality of workers by using test questions (whose answers are known before-hand) to discard the answers of workers who do not substantially pass the test questions. Moreover, the system also supports peer review and provides statistical reports on the outcome of worker answers, for the purposes of evaluating and profiling workers.

2.4.2 Human-powered Applications

Web and mobile technologies have enabled massive collaboration across the world, especially with the emergence of real-time human-empowered applications [HVT+18]. For instance, humans act as sensors to help machines monitor public health at scale via personal smartphones [CTL+17] (aka participatory sensing). Mobile Millennium project is another example that uses GPS-enabled mobile phones to collect en route traffic information and upload it to a server in real-time [TTFS18]. Recently, real-time food ordering and delivery services have been enabled by spatial crowdsourcing, in which crowd workers service spatially located requests through their smartphones [DGD+17]. Human experts are always needed to tune capacity, universality, and quality of AI models [LZZ+18]. Motivated by these successes, this project utilizes human validators to aid algorithmic models in debunking misinformation.

Human-powered machine learning. In many machine learning applications (e.g., classification, visual recognition, and object detection), training data is needed to parameterize the automatic models. However, in traditional systems, training data is often limited (e.g., only a single expert is hired) and out of date (e.g., old data is used several times for modern techniques). To address this problem, a large body of works [SRYD14, BWS+10, VG14] employs human computation in the form of crowdsourcing to iteratively improve the training data. All of these works are feasible due to the mass availability of thousands of crowd workers and their cheap hiring cost in online platforms (e.g., as stated earlier, Amazon Mechanical Turk, CrowdFlower).

In general, these works design an active learning process, which is executed as follows. Firstly, automatic results are produced based on existing training data. Then, crowd workers are employed to validate these results. Combining worker inputs, the system generates new training data. On top of the new training data, another iteration is performed using the same automatic techniques. In brief, this active learning process entails a human computation loop, in which human users iteratively contribute to improve the output quality of automatic tools.

Participatory sensing and measuring. In participatory sensing, people are equipped with built-in sensors in their smartphones to measure real world physical conditions. There are various applications in this category such as environmental monitoring and tracking daily activities.

Traditional environmental monitoring systems rely on aggregate statistics by fixed sensors to measure and report environmental pollution over an area (e.g., community, city, state). These systems have several limitations. For example, since the cost of sensor deployment is high, the sensors might not cover all the desired regions. Moreover, due to various factors including low battery or damaged components, the deployed sensors might report imprecise measurement. To overcome these limitations, participatory sensing systems (e.g. Common Sense [DAK+09]) use specialized handheld air quality sensing devices, which are distributed to a large number of human participants, for collectively

measuring various air quality indices such as carbon emissions, noise levels, and water conditions.

Similar methods are demonstrated in PEIR – another participatory sensing system [MRS+09], which uses mobile phones to infer daily activities of human participants. For example, the combination of personalized accelerometer data and a sequence of locations from the GPS can recognize the transportation mode of a user, such as biking, driving, or taking public transports (bus, metro).

Human-powered data curation. Annotation is the process of attaching metadata (e.g., comments, tags, markups) to different types of data such as images, text, media, etc. Since data has different characteristics and formats, automatic annotation tools might not be able to produce meaningful annotations that satisfy user needs. Moreover, humans often understand the content of data more easily than computers (e.g., watching videos). As such, many research works employ human computation for the purposes of data annotation. As an example, the authors of [KNW+14] designed a system that enables human participants to annotate step-by-step structure for an existing video, in which each step is a meaningful segment with textual and visual annotations of the video content. Human annotations are then combined by majority voting to decide the best annotations for the videos.

Human-powered Trust Management. Information credibility has been studied in the literature to evaluate the trustworthiness of a data source or an artifact, indicating whether the information is trusted or not. With the growth of the web, information credibility is applied to assess the credibility of websites. However, assessing the credibility of information on the web is still a challenging issue. As a publicly available platform in which anyone can share anything, the web is inherently uncertain, in which information published cannot be easily verified for validity, legitimacy and trustworthiness. Moreover, as the web contents are shared by humans, automatic techniques might not be able to truly assess the credibility of web content.

Overcoming this issue, a large body of work employs human computation to evaluate the credibility of websites. In general, these works collect users' feedback by allowing them to provide ratings on a web page. Possible rating scores can be binary (e.g., Postive/Negative [Giu10]) or multiple (Trustfulness/Unbiased/Security/Page design [HOA13]). The ratings are then combined to produce the credibility score of the web page (e.g., by computing the means and standard variances of user rating scores [Giu10]).

2.4.3 Feedback Guidance

While there are various optimisation issues of human-powered validation such as quality control [HTTA13], workflow control [MGAM16], latency control and task design [LWZF16, CCAY16], we focus on a cost control problem, i.e. effort minimisation, to guide user in validating misinformation flags.

User guidance. Guiding users has been studied in data integration, data repair, crowdsourcing, and recommender systems [JFH08, YEN+11, LSD+17]. Most approaches rely

on decision theoretic frameworks to rank candidate data for validation. Despite some similarities in the applied models, however, our approach differs from these approaches in several ways. Unlike existing work that focuses on structured data that is deterministic and traceable, we cope with Web data that is unreliable and potentially nondeterministic. Also, instead of relying on two main sources of information (data and data provider), we incorporate individual features as well as direct and indirect relations between data types (sources, documents, claims).

Our setting is also different from active learning, as we do not require any training data for a user to begin the validation process. Moreover, we incrementally incorporate user input without devising a model from scratch upon receiving new labels. However, stopping criteria for feedback processes have been proposed in active learning, e.g. using held-out labels [MSF+14] and performance estimation [LS08]. Yet, these methods are applicable only for specific classifiers and do not incorporate human factors. Using our probabilistic model, we have been able to propose several criteria for early termination that turned out to be effective in our experimental evaluation.

Moreover, we focus on reducing manual effort, assuming that there is a notion of truth. Yet, user input may be uncertain or subjective [AGMS13]. While we consider the integration of such feedback to be future work, we see two scenarios with different implications. First, if claims are validated by a single biased expert [STVB16], the grounding function is shifted to the expert belief. This angle can be extended to recommender systems, which recommend the most belief-compatible claim for a user. Second, if claims are validated by multiple biased experts, differences in their belief suddenly have an impact. Finding a common ground then requires negotiation and conflict resolution mechanisms [GBKS09].

Active learning. Active learning approaches allows the learning algorithm to choose the data it wants to get labels [Set09], which is suitable in settings where the labels are costly to obtain. It has been applied in various machine learning applications such as information fusion [PGQdC17, PDA17], speech recognition [ZLR05], information extraction [AED99] and text classification [TK02]. In credibility assessment, active learning approaches can be classified into two categories: classifier-independent and classifier-specific. Classifier-specific approaches requires constructing a classifier. Notable works in this category are SVMs [TK02] and decision trees [ZE01] while committee-based approaches [AED99] belong to classifier-independent category. Although our approach is similar to active learning approaches, there is a fundamental difference. Traditional active learning approaches do not guarantee that the values returned by the learning algorithm satisfy a predefined quality. Moreover, they still require a lot of training data, which is costly, and might not reach the required quality [Set12]. On the other hand, our approach is able to achieve this constraint while maximizing the output size.

Implicit Feedback. Asking users directly for validation might incur some overhead time to setup the procedure and incentive mechanism. For this reason, different frameworks are established to collect implicit feedback from users via their social posts, due to the fact that users might write an assertion post with evidences to confirm or reject

a story [VR15]. Similar ideas of constructing evidences and supporting information for a rumour are also explored, e.g. by collecting all preceding social posts in the same event [ZLP17]. However, such approaches might introduce addition errors coming from the preprocessing step, e.g. to extract evidences from textual data. Moreover, the collected evidences themselves do not have a high confidence since the validating users are not necessarily qualified to be experts. Further steps for credibility assessment of validating users might be a solution, but implying a "chicken or egg" dilemma where a large amount of historical data needs to be collected for references. To the best of our knowledge, such a referencing dataset is still limited (e.g. Snopes) due to the fact that different types of misinformation stories behave very differently [VRA18], making it difficult to generalise from historical records.

2.5 Data Visual Analytics

2.5.1 Social data visualisation

Social data visualisation or visual analytics is often based on topic modelling [BNJ03], feature extraction [MWDNM14b, WP15, NDN+17], and temporal-aware information processing [CTY+16]. Methods for topic modelling, e.g., Latent Dirichlet Allocation [BNJ03], hierarchical Dirichlet processes [GT04], or word modelling [ZBG13], are not applicable for a streaming setting, since they require multiple passes over the data. Streaming versions of these techniques [HBB10, CSG09], in turn, ignore the dynamics of social data. Our model follows a non-parametric approach, where the number of topics and vocabulary words is learned from the data rather than specified in advance. Moreover, our model incorporates social features [MWDNM14b, EMSW14] when assessing data utility. Also, methods to query a streams of social data [KDM16, YLC16] are not applicable for data summarization, since the query is not known in advance.

2.5.2 Streaming data management

Data stream management systems (DSMS) handle a continuous flow of incoming data records, such as postings in social networks. In the Big Data era, as it is necessary to process a large amount of data in a very short time, in-memory DSMS have been developed such as Storm, Yahoo! S4, Spark Streaming, and MapReduce Online. Several indexing mechanisms are also proposed such as textual indexing [WHMO13] with a composite tree structure and a shared list of stream events to speed up response time. On top of the state-of-the-art, this thesis will propose techniques to retain important information from social streams, in particular textual data. To achieve a fast response with minimal overheads in a memory-efficient manner, an information sketch is incrementally updated upon the arrival of new data to summarise historical data without storing all of them.

2.5.3 Data summarisation

Traditional data summarisation works on offline data [ZRH+16, NMD12, NNWA15, HTTA13] and, even if temporal aspects are considered [CTY+16, ABR16, CA13], on the whole data. Existing streaming algorithms for data summarization also rely on access to the complete data, as they sample the data for an estimation of the utility [MBK+15, MKSK13, BMKK14]. A relaxed version of the retaining problem has been addressed in [ELVZ16], which finds subsets of items with maximal utility using a sliding window. Yet, different from our problem formulation, this method targets solely the recent data bounded by a fixed window size, discarding all old items. Whereas, our approach retains old items as long as they are valuable. Also, unlike [ELVZ16], our approach summarizes the entire history of a data stream. Finally, the algorithm in [ELVZ16] assumes apriori knowledge of an upper bound of utility. While this assumption may be reasonable for some types of data streams, it it unrealistic for dynamic data produced by social platforms.

2.5.4 Exploratory Visual Analytics

Our problem setting is similar to the one of database exploration techniques [IPC15], especially in the context of multi-objective optimisation problems [Deb01]. That is, users cannot formulate their interests as a query until the data of interest is shown to them. Data exploration scenarios with a single goal often employ an *information sketch* or *histogram* to approximate the distribution of data [Ioa03, ESC16, GSW04, JKM+98, GKS01]. Different types of histograms have been studied in the literature, such as: (i) equi-width histograms [MZ11], (ii) equi-depth histograms [MZ11], (iii) v-optimal histograms [ILR12], (iv) maxdiff-histograms [PHIS96], (v) compressed histograms [PHIS96]. The quality of histogram construction is measured by error functions that are specific to application domains and data characteristics. Examples include the sum of squares of absolute errors [WSA15], maximum error metrics, and relative error metrics [Ioa03].

Histograms have also been studied in signal and image processing [JS94] under the name of *wavelets*. Here, the idea is to apply hierarchical decomposition functions to transform raw data into wavelet coefficients. Despite having different concepts and techniques, wavelet construction shares some similar complexity and quality results with histogram construction.

The state-of-the-art in histogram construction is limited to a single outcome dimension. So-called multi-dimensional histograms [LKC99, PI97] are either based on dimensionality reduction (SVD and Hilbert numbering) and lack guarantees on the result quality, or partition each dimension incrementally (MHIST [PI97]). In the latter case, a dimension is partitioned only based on the partition of the previous dimension, which neglects any trade-off between outcome dimensions.

Chapter 3

Misinformation Detection: The Case of Rumour Early-Detection

From Anomaly Detection to Rumour Detection using Data Streams of Social Platforms

VLDB 2019

Social platforms became a major source of rumours. While rumours can have severe real-world implications, their detection is notoriously hard: Content on social platforms is short and lacks semantics; it spreads quickly through a dynamically evolving network; and without considering the context of content, it may be impossible to arrive at a truthful interpretation. Traditional approaches to rumour detection, however, exploit solely a single content modality, e.g., social media posts, which limits their detection accuracy. In this chapter, we cope with the aforementioned challenges by means of a multi-modal approach to rumour detection that identifies anomalies in both, the entities (e.g., users, posts, and hashtags) of a social platform and their relations. Based on local anomalies, we show how to detect rumours at the network level, following a graph-based scan approach. In addition, we propose incremental methods, which enable us to detect rumours using streaming data of social platforms. We illustrate the effectiveness and efficiency of our approach with a real-world dataset of 4M tweets with more than 1000 rumours.

3.1 Introduction

Social platforms became widely popular as a means for users to share content and interact with other people. Due to their distributed and decentralised nature, content on social platforms is propagated without any type of moderation and may thus contain incorrect information. Wide and rapid propagation of such incorrect information quickly leads to *rumours* that may have a profound real-world impact. For instance, in April 2013, there was rumour about two explosions in the White House, injuring also Barrack

Obama [ZRM15]. The rumour was fuelled by content posted using a hacked Twitter account associated with a major new agency. The resulting panic had major economic consequences, such as a $136.5 billion loss at the stock market. This incident highlights the need for early and accurate *rumour detection*, in particular on social platforms.

It is notoriously hard to detect rumours [VRA18]. Posts on social platforms are short and lack semantics. For instance, tweets have a limited number of characters, and comprise slang and spelling mistakes. Hence, traditional techniques to assess the credibility of (long, well-written) documents are of limited use for social platforms. Also, user interactions at unprecedented scale lead to rumours spreading quickly. Earliness of rumour detection is as important as detection accuracy. Moreover, social platforms are dynamic. Content is posted continuously, so that rumour detection cannot exhaustively collect data before giving results, but needs to work with streaming data. Finally, posts on social platforms are contextual. A post in isolation may not provide sufficient information for rumour detection. Instead, modalities such as user backgrounds, hashtags, cross-references, and user interactions must be considered to improve detection accuracy.

Several debunking services such as snopes.com have been established to expose rumours and misinformation. They harness collaborative user efforts to identify potential rumours, which are then verified by experts. Due to such manual processing, the number of potential rumours that can be assessed is limited and significant time is needed for verification, which motivated work on automated rumour detection. Given the short length of posts on social platforms, rumour detection is often approached by grouping posts that relate to a single event [MGM+16]. This does not work in an online setting, though, since the posts related to an event are not available a priori.

Traditional rumour detection techniques tend to rely solely on the textual information of posts, potentially combined with features on post authors and their relations. However, focusing on one or two modalities of posts on social platforms is insufficient. For instance, users posting rumour-related content are often ignored by other users, which is not directly visible in features that capture solely the characteristics of a single user. In another example, posts circulating among a group of users that believe in conspiracy theories are likely to refer to rumours. Without information from outside the group, it is impossible to know whether these posts are related to a rumour.

Our approach. Against this background, we argue for a novel approach to rumour detection that identifies anomalies on social platforms by comparing data *between peers* and *with the past*. Such anomalies can be observed for different modalities (e.g., users, tweets) and at varying levels of granularity. For example, a sudden increase or decrease in the number of followers of a user may be related to the user spreading rumours. Also, within a group of users, the credibility of one user being significantly lower than their peers may stem from the propagation of rumours. Moreover, relations between entities (e.g., users, posts, hashtags, links) may hint at anomalies, e.g., differences in time and location mentioned in a tweet and in a linked article.

In this chapter, we present models and methods to realise the idea of detecting rumours based on anomalies. To this end, we follow a data management approach: We

ground rumour detection in algorithms that work on a generic graph representation of social data, thereby achieving a solution that is applicable for any type of social platform. We first show how to identify anomalies locally, by assessing entities and relations of a social platform in comparison to their peers and to their past. Yet, acknowledging the inherent randomness of social platforms, anomalies are then viewed at a broader scale. To conclude on the spread of rumours, which is deemed more important than their classification [VRA18], we incorporate the vicinity of local anomalies.

Our contributions and the structure of the chapter (following a discussion of some background in Section 3.2) are summarised as follows:

- *Social Platform Model and Rumour Detection (Section 3.3).* Based on a model for social platforms, we develop a general process to detect rumours based on local and global anomalies.
- *Local Anomaly Detection (Section 3.4).* We propose a non-parametric method for anomaly detection at the level of individual entities, based on differences between (i) current and past observations related to an entity, and (ii) the entity and its peers.
- *Global Anomaly Detection (Section 3.5).* We lift anomaly detection to groups of entities, taking into account relations between them.
- *Streaming Setting (Section 3.6).* We show how to apply our approach for streaming data by incrementally computing anomaly scores on the local and global level.

An evaluation of our approach with more than 4M real-world tweets, spanning more than 1000 rumours, is presented in Section 5.5. We conclude the chapter in Section 5.6.

3.2 Motivating Example

Anomalies in social media. Abnormal propagation of information on social platforms can be classified as different types of anomalies, including hypes, fake news, satire news, disinformation, misinformation, and rumours [ZAB+18]. For hypes, information is propagated in cascades that accidentally 'blow-up' on social platforms, e.g., related to popular events. Rumours, in turn, originate from the fact that people tend to exaggerate what they dislike [Ver]. Their veracity needs to be assessed, which is commonly done by assigning a trust score to entities, such as users and posts [Eng].

Here, we focus on detecting rumours. While hypes and rumours share some characteristics, they differ in how information is propagated. In hypes, information is spread randomly and chaotically. As revealed in a recent survey [VRA18], however, rumours are propagated in a channelled manner, spreading 'farther, faster, and deeper' through interactions of actual users rather than bot accounts.

Type of anomalies differ in their sets of indicative signals. For example, detection of hypes (e.g., breaking news) focuses on peak volume of social posts and sharing activities [OCDA15, OCDV14]. Spam detection of online reviews, in turn, uses user signals, such as average rating, number of reviews, and selectivity [YKA16]. Our approach for rumour detection looks at inconsistency signals, exemplified below.

Twitter as an example. While we use Twitter as an example of a social platform throughout the chapter, our model is applicable to other social platforms [SLZ+17], as it is based on a universal graph representation (Section 3.3), generic statistical measures to compute anomalies (Section 3.4), and a graph-based anomaly detection algorithm (Section 3.5).

Consider a snapshot of Twitter social graph, as shown in Figure 3.1. It includes users, tweets, hashtags, and linked articles. Each entity has different features, e.g., a user has a registration date and a number of followers. Entities are connected by relations. For instance, the relation between a tweet and an article indicates that the content of the tweet contains a link to that article. Moreover, each relation has an attribute value, e.g., the tweet-article relation has an attribute that indicates the difference between the publication dates of the tweet and the article, respectively.

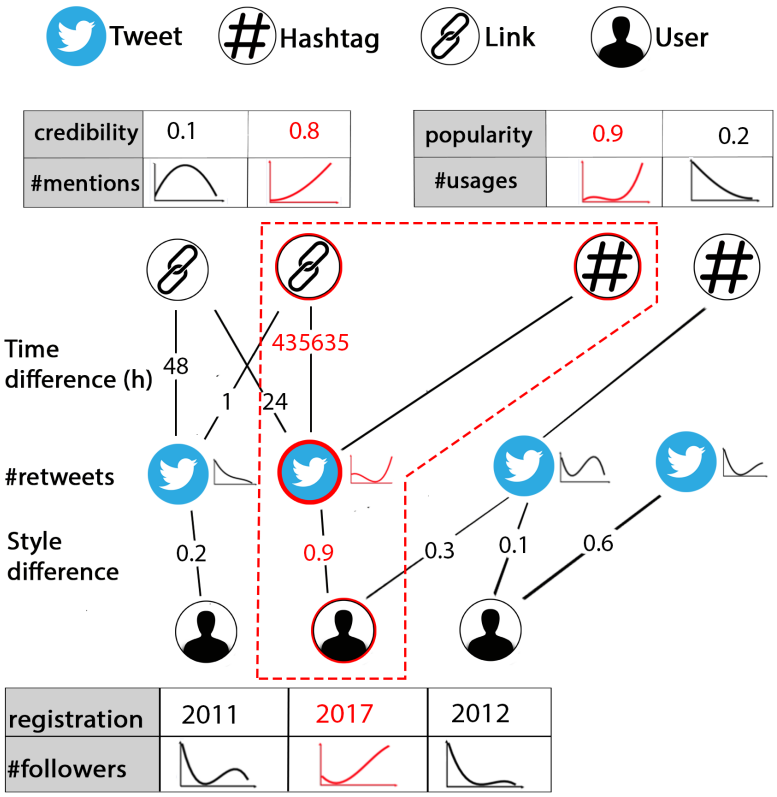


Figure 3.1: Multi-modal social graph

Rumours are often manifested in anomalies related to entities and their relations. In Figure 3.1, one may observe that the highlighted user has a registration date that is significantly newer than those of related users. At the same time, the number of followers is very high, compared to the historical record of the user. Other entities in this example are also suspicious, due to anomalies. For the highlighted tweet, the number of retweets is suddenly higher than in the past, as is the number of mentions for the highlighted linked article.

The above local anomalies provide a first signal for rumour detection. Yet, in isolation, these signals are not reliable. For instance, a user sparking a hype will also

experience a sudden increase in the number of followers. We therefore need to consider *global anomalies* that comprise connected entities for which local anomalies have been observed. In the example, a rumour-related user is expected to post a rumour-related tweet, which links to a rumour-related article. Moreover, these connections between entities are also meaningful for rumour detection. For instance, in Figure 3.1, the time difference between the highlighted tweet and linked article is suspicious, as is the difference between the regular linguistic style of this user (derived from past tweets) and the style of this particular tweet.

In this work, we provide the methods to realise the above idea: We exploit local anomalies and, based thereon, global anomalies among the entities of a social platform to reliably detect rumours.

3.3 Model and Approach

Below, we present a model to capture entities of a social platform and their relations (Section 3.3.1). We then define the rumour detection problem (Section 5.2) and outline our approach to address it (Section 3.3.3).

3.3.1 A Model of Social Platforms

A social platform comprises many entities that are linked to each other by relations.

Entities (nodes). Our model comprises entities of specific types, i.e., modalities, such as tweets, links, users, and hashtags. Entities are modelled using feature vectors, where the features depend on the entity type. For the example in Figure 3.1, each user has registration date and number of followers as features. While we limit the discussion to the above modalities in the remainder of this chapter, our model is generic in the sense that further modalities such as images and videos [KLPM10] can be incorporated.

Relations (edges). Characteristics of entities in isolation are not sufficient to detect rumours. The relations between them provide a richer picture and thus can be expected to be beneficial for rumour detection. Each relation is also modelled by a feature vector, which is specific to the the type (or modality) of the relation. For the example in Figure 3.1, each tweet-article relation has the time difference between the publication times of tweets and linked articles.

Multi-modal social graph. A *multi-modal social graph*, or *social graph*, is composed of modalities, entities, and relations between entities. We denote by $D = \{D_1, \ldots, D_n\}$ a set of entity types, while $V = V_1 \cup \ldots \cup V_n$ is a set of entities, such that V_i is the set of entities of type D_i. Similarly, $C \subseteq [D]^2 = \{C_1, \ldots, C_m\}$ is a set of relation types ($[D]^2$ being the 2-element subsets of D), $E = E_1 \cup \ldots \cup E_m$ are sets of relations, where E_i is the set of relations of type C_i.

Based thereon, a social graph is defined as $G = (Q, V, E, f)$, where $Q = D \cup C$ is called the set of modalities of G. The feature information f of entities and relations is used to capture rumour signals in a social graph. Formally, $f = \{f_1, \ldots, f_{n+m}\}$ is a set

of mapping functions, where $f_i : Q_i \to \mathbb{R}^{q_i}$ defines an q_i-dimensional feature vector $f_i(x)$ for each element x of the modality Q_i.

The notion of a social graph enables us to address rumour detection with techniques for data management. As such, the developed algorithms are also applicable to data of social platforms that can be transformed to a graph representation [ZZZ+17, TF13, JLLH11, SAH12].

3.3.2 Rumour Detection

In a social graph, rumours materialise for a subset of its entities. The definition of this subset is not known, so that its identification is referred to as the rumour detection problem. That is, there is some (unknown) function that assigns truth values to entities (regular or rumourous), which shall be approximated.

Problem 1. *Given a social graph $G = (Q, V, E, f)$ and a ground-truth set $R^* \subseteq Q$, the* rumour detection problem *is to find a label function $l : Q \to \{1, 0\}$ to categorize which entities are rumourous, such that detection coefficient is maximized:*

$$\frac{|R^* \cap R|}{|R^* \cup R|} \quad \textit{with } R = \{x \in Q \mid l(x) = 1\}.$$

While the above definition is independent of the type of entity that is considered rumourous, in the remainder, we focus on the detection of rumourous tweets. The reason being that there is no clear-cut truth function to label other entities. For example, users may spread rumours in some tweets, but propagate regular information in others.

3.3.3 Approach Overview

Addressing the above problem requires us to overcome the trade-off between accuracy and completeness, which is difficult [BG94]. A common strategy is to first focus on completeness and subsequently optimize the accuracy of rumour detection. Filtering out false positives is often easier than finding additional true positives.

Following this line, we first strive for completeness by collecting all rumourous signals in data features: The more anomalous a feature of a tweet, the more rumourous it is. However, such a feature-based approach alone will not yield high accuracy of rumour detection. Since there is always randomness and noise in the data of a social platform, we conclude that a tweet is rumourous only if it is part of a rumourous graph structure. For example, in Figure 3.1, the highlighted subgraph denotes such a structure for the respective tweet, capturing rumourous context related to a user, hashtag, and linked article.

Retrieving all rumour signals from a social graph, we then reduce false positives by cross-checking between the signals, while incorporating their contexts. More precisely, we use the structural information of a social graph (i.e. relations between entities) to find a subgraph that is most rumourous. The tweets contained in this subgraph are then considered to be the actual rumour.

Rationale. Our approach is driven by the following observations:

- Identifying solely individual rumourous tweets ignores the rumour structure, i.e., it neglects that a cluster of rumourous tweets denotes a single rumour. Hence, rumour detection shall incorporate the co-occurrence of rumourous tweets as part of a rumour.
- Identifying rumours solely on the level of tweets neglects the interplay of modalities in rumour propagation. A social graph defines complex relations between entities, so that the identification of rumourous tweets, e.g., leads to the identification of rumourous users, hashtags, and links. Hence, the structure of a social graph shall be exploited to assess the propagation of rumourous information. This way, the need to detect explicit events by aggregating entities is eliminated, which is a common first step in traditional rumour detection [PBKL14].

Framework. Against this background, we design a two-step rumour detection process, illustrated in Figure 3.2. In a first step, we aim to detect local anomalies in entities and relations. In a second step, these local anomalies and the relations in the graph enable the detection of rumours at the subgraph level. Below, we summarise the two steps, while their details are given in Section 3.4 and Section 3.5, respectively.

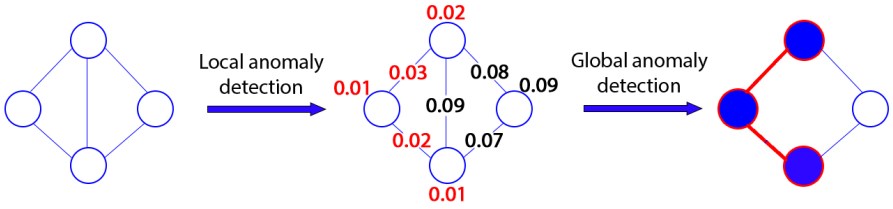


Figure 3.2: Rumour as Anomaly Detection Process

Local anomaly detection. First, we design a function that assigns an anomaly score to each entity. We argue that an anomaly scoring shall satisfy the following requirements:

(R1) *Completeness:* In order to eliminate false negatives in rumour detection, the identification of anomalies in the data shall be comprehensive. That is, complementary angles to identify deviations from expected observations should be considered.

(R2) *Uniformity:* For entities of all modalities, there shall be a uniform scoring domain (independent of the number of features), with a uniform ordering (lower value indicating more rumourousness), and a uniform distribution (scores are uniformly distributed in $[0, 1]$). The latter is important as thresholding for rumour detection is challenging for non-uniform distributions.

(R3) *Non-parametric:* We assume that features follow an unknown baseline distribution. It is estimated based on the data and serves to assess the level of anomalousness per entity.

Global anomaly detection. Second, we rely on the detected local anomalies and aim at the detection of global anomalies, which indicate rumours. This shall incorporate the following requirements:

(R4) *Cross-checking:* In order to avoid false positives, rumourousness between neighbouring entities shall be cross-checked in the social graph. As content on social

platforms is dynamic and rumours may propagate very quickly, a rumourous entity is expected to affect its neighbours immediately. Hence, global anomaly detection shall consider the context of local anomalies.

(R5) *Structuredness:* Any algorithmic solution to detect global anomalies shall acknowledge the structure of rumours. The 'rumour-related' parts of a social graph, in terms of rumourous information that jointly denotes a rumour, shall be detected.

(R6) *Non-parametric:* The scoring of a global anomaly shall not assume any prior distribution of local anomaly scores. This supports multi-modality and robustness to different datasets.

3.4 Local Anomaly Detection

This section is devoted to the computation of local anomaly scores in a social graph. Guided by the above requirements (R1, R2, R3), we first show how to construct features for identifying rumours (Section 3.4.1). Then, we introduce history-based anomaly scores (Section 3.4.2) and similarity-based anomaly scores (Section 3.4.3). Based thereon, a unified anomaly score is derived for each graph element (Section 3.4.4).

3.4.1 Features to Identify Rumours

Feature engineering is the only domain-specific step of our approach, which we illustrate here for the case of Twitter. We distinguish history-based and similarity-based features. The former capture differences between the current and past state of an entity. The latter help to cross-check the differences between entities and relations of the same type. Specifically, we consider the following features per modality, see also Table 3.1:

- User: The *registration age* and *credibility score* are considered indicators for rumours, since users spreading rumours tend to create new accounts to hide their identity. Moreover, sudden changes in the *frequency* of status updates, the number of *followers*, and the number of *#friends* may be related to rumours.

- Tweet: We consider *keywords* and the *linguistic style*. Tweets that are subjective or emotional are more likely to be rumour-related as they aim to provoke strong emotions to promote sharing. Also, the number of *retweets* may indicate rumours.

- Link: Articles linked in tweets may indicate rumours, which we assess based on the *credibility score* and *linguistic style* of the linked source and article, respectively. Furthermore, the number of *mentions* over time is used as a feature.

- Hashtag: The *popularity*, as measured by a semantic ranking [BJV15], and sudden changes in the number of *usages* of a hashtag are expected to be rumour-related.

We further consider the features of relations between entities:

- Tweet-Link: The *time*, *location*, and *event* mentioned in a tweet may be different from the respective details given in the linked article. Also, the linguistic *style* of the tweet may be different from the one of the linked article.
- User-Tweet: The linguistic *style* of a tweet may differ from the regular style of the user who posts it.
- User-Link: The *source* linked in a tweet is anomalous.
- User-Hashtag: The hashtag is *novel*, i.e., it has not been used by the user before.
- Link-Hashtag: The hashtag has been *mentioned* in the linked article with an anomalous frequency.

While some of the features are static (similarity-based), others are dynamic (history-based), so that they are derived from time snapshots using streaming APIs, such as [MPLC13]. We compute the features using established methods, whose details are described in the experiment section (Section 3.7.2).

Table 3.1: Features to identify local anomalies.

	Element	**Feature**	**Anomaly Type**
Entities	User	registration age	similarity-based
		credibility score	similarity-based
		status frequency	history-based
		#followers	history-based
		#friends	history-based
		#tweets	history-based
	Tweet	keywords	similarity-based
		linguistic style	similarity-based
		#retweet	history-based
	Link	credibility score	similarity-based
		linguistic style	similarity-based
		#mentions	history-based
	Hashtag	popularity score	similarity-based
		#usages	history-based
Relations	Tweet-Link	time	history-based
		location	similarity-based
		event	similarity-based
		style	similarity-based
	User-Tweet	linguistic style	similarity-based
	User-Link	source	similarity-based
	User-Hashtag	novelty	similarity-based
	Link-Hashtag	mentioning	similarity-based

Using the above features independently may lead to false positives. For instance, although rumours usually have a specific linguistic style, the reverse is not always true as, e.g., news about tragedies also adopt an emotional style. To mitigate such effects, we consider the above diverse set of features, which addresses requirement R1.

3.4.2 History-based Scoring

An anomaly score may be based on the differences between the current and past values of a feature vector. To this end, we establish a baseline distribution for each attribute to represents the normal behaviour, in the absence of any rumour. Then, based on the baseline distribution and the current feature values, we estimate an empirical p-value to measure the anomalousness of a feature. Aggregating these values, we asses the anomalousness of an entity or relation.

Deriving historic data. To derive historic values of features of entities or relations, we apply a temporal window. For an entity or relation x, the historic data is denoted by $X_t = \{x_1, \ldots, x_t\}$, where all x_i are temporal snapshots of x. This way, historic data of the same length is considered for different history-based features of x, which enables the integration of features with varying temporal properties. Yet, t is not fixed across entities or relations, so that historic data of different lengths may be incorporated for different modalities. Note that collecting historic data is straight-forward for common platforms. Details on our data collection can be found in Section 3.7.2.

Anomaly score of a history-based feature. Our computation is based on the following null hypothesis: If there is no rumour and we select a random observation from the past, how likely is it that its value is greater than or equal the current one? Based on historic data, the anomaly score of a feature $j \in [1, q_i]$ of an element (entity or relation) $x \in Q_i$ at timestamp t is defined as the statistical confidence degree (i.e., the p-value, the lower the better) [CN14]:

$$p_T(f_{i,j}(x_t)) = \frac{|\{x_r \in X_{t-1} : f_{i,j}(x_r) \geq f_{i,j}(x_t)\}|}{|X_{t-1}|} \tag{3.1}$$

where $f_{i,j}(x_t)$ refers to the j-th component of the feature vector $f_i(x_t)$ of an element x at timestamp t. In other words, the p-value is computed based on the number of past values $f_{i,j}(x_r)$ that are greater than the current observation $f_{i,j}(x_t)$. This is a *non-parametric* statistical measure (addressing requirement R3), since it does not assume any prior distribution on the historic data.

Example 1. *Consider a Twitter user @jacobawohl (x), who is related to rumours about the Las Vegas shooting in 2017 [Snoa]. The number of active followers (feature 1) and the number of tweets (feature 2) of the user at three consecutive time points is* $\{4.72K, 294, 7.03K\}$ *and* $\{102, 43, 51\}$ *respectively. At the third time point, the p-values of feature 1 and feature 2 are* $p(f_1(x_3)) = \frac{0}{2} = 0$ *and* $p(f_2(x_3)) = \frac{1}{2} = 0.5$*. At the second time point, these values are* $p(f_1(x_2)) = \frac{1}{1} = 1$ *and* $p(f_2(x_2)) = \frac{1}{1} = 1$*. Moreover, at the first time point, there is no historic value and we set* $p(f_1(x_1)) = p(f_2(x_1)) = 1$*.*

History-based anomaly score. The non-parametric p-value of an entity or relation x specifies its anomaly score based on historic observations. We aggregate these anomaly scores as follows [CN14]:

$$p_T(x_t) = \frac{|\{x_r \in X_{t-1} : p_{min}(x_r) \leq p_{min}(x_t)\}|}{|X_{t-1}|} \tag{3.2}$$

where $p_{min}(x_r) = min_{j=1...q_i} p(f_{i,j}(x_r))$. That is, at each timestamp, we compute the minimum value over all features. Then, the anomaly score $p_T(x_t)$ is the number of past minimum feature values $p_{min}(x_r)$ that are less than the current minimum feature value $p_{min}(x_t)$.

The reason for using *min* for the aggregation is to avoid false negatives, where some features are anomaly-significant, whereas others are not. Moreover, we do not consider the minimum p-value over all features at a single timestamp directly, since elements can have different numbers of features. Rather, our idea is to cross-check the scores between different timestamps across features, so that our aggregation yields *uniform* scores over all entities and relations, regardless of their modality, which addresses requirement R2.

Example 2. *Taking up Example 1, we derive that* $p_{min}(x_1) = \min\{p(f_1(x_1)), p(f_2(x_1))\} = 1$ *as well as* $p_{min}(x_2) = \min\{p(f_1(x_2)), p(f_2(x_2))\} = 1$, *and* $p_{min}(x_3) = \min\{p(f_1(x_3)), p(f_2(x_3))\} = 0$. *The p-value of user x at the current timestamp is* $p(x_3) = (0+0)/2 = 0$. *With a confidence level of 99%, we say that the user is involved in some rumour, since* $p(x_3) \leq 0.01$.

3.4.3 Similarity-based Scoring

Anomalousness can also be quantified by differences between entities and relations of the same type. For instance, the linguistic style of a tweet is a static property, that often lacks historic data, but may be a strong indicator of rumours. We therefore establish a baseline for features of static properties, as detailed below.

Anomaly score of a similarity-based feature. The null hypothesis of this case is summarised as: If there is no rumour, how likely does a randomly selected set of observations for a feature of different elements (entities or relations) of the same modality would have values greater than the considered element. We capture the null distribution of a feature of an element x of modality Q_i using the feature values of its peers ($x' \in Q_i$). Then, the p-value of a similarity-based feature $j = 1 \dots q_i$ of an element x is defined as follows:

$$p_S(f_{i,j}(x)) = \frac{|x' \in Q_i : f_{i,j}(x') \geq f_{i,j}(x)|}{|Q_i|} \tag{3.3}$$

That is, the p-value is computed based on the number of values $f_{i,j}(x')$ from other elements of the same modality that are greater than the value of the current element, $f_{i,j}(x)$. This p-value is also non-parametric (as defined by requirement R3), since it does not assume any prior distribution on the elements.

Example 3. *Now, consider three Twitter users @prisonplanet (x), @wes_chu (y), @jacobawohl (z), who have registration ages (feature 1) of {8, 6, 1} and average credibility scores (feature 2) of {-5, -4, -3} (0 means least credible). For feature 1, we have* $p(f_1(x)) = 1$, $p(f_1(y)) = 2/3$, $p(f_1(z)) = 1/3$. *For feature 2, we have* $p(f_2(x)) = 1$, $p(f_2(y)) = 2/3$, $p(f_2(x)) = 1/3$.

Similarity-based anomaly score. Again, based on the p-value of a similarity-based feature of an element x, the similarity-based anomaly score of x is defined as follows:

$$p_S(x \in Q_i) = \frac{|x' \in Q_i : p_{min}(x') \leq p_{min}(x)|}{|Q_i|} \tag{3.4}$$

where $p_{min}(x') = \min_{j=1...q_i} p_S(f_{i,j}(x'))$. For each element, we compute the minimum value over all features. Then, the anomaly score of an element is the number of elements such that the minimum feature value of the current element is larger than their minimum feature values. As above, we choose *min* as an aggregation function to avoid outliers. We also aggregate across elements rather than features of a single element only. This yields *uniform* anomaly scores of elements from different modalities (requirement R2).

Example 4. *We continue with Example 3 and derive* $p_{min}(x) = \min\{p(f_1(x)), p(f_2(x))\} = 1$, $p_{min}(y) = \min\{p(f_1(y)), p(f_2(y))\} = 2/3$, *and* $p_{min}(z) = \min\{p(f_1(z)), p(f_2(z))\} = 1/3$. *The p-value of* z *is* $p(z) = (0 + 0 + 1)/3 = 0.33$. *With a confidence level of 65%, we say that user* z *is involved in some rumour, since* $p(z) \leq 0.35$.

3.4.4 Unified Scoring

As both entities and relations show history-based and similarity-based features, we combine the respective anomaly scores:

$$p(x) = \min\{p_T(x), p_S(x)\} \tag{3.5}$$

where $p_T(x) = 1$, if x has no history-based features, and $p_S(x) = 1$, if x has no similarity-based features. Again, *min* is used in the aggregation to avoid outliers.

We note that $p_T(.)$ and $p_S(.)$ are uniformly distributed in $[0, 1]$ under the assumption that, in the absence of rumours, (i) the current observations are interchangeable with observations in the past; and (ii) the current observations of an element are interchangeable with observations from other elements. Based thereon, the probability that $f_{i,j}(x_r) \geq f_{i,j}(x)$ and $f_{i,j}(x') \geq f_{i,j}(x)$ is 0.5, which makes $p_T(f_{i,j}(x))$ and $p_S(f_{i,j}(x))$ follow a uniform distribution in $[0, 1]$. Also, the minimum of p-values from different features are interchangeable with past minimum values or from other peers, so that $p_T(x)$ and $p_S(x)$ are uniformly distributed in $[0, 1]$.

The *uniform* distribution of p-values is important: It enables us to handle the heterogeneity of a social graph, as different elements and modalities are mapped to the same domain of p-values. Moreover, the model facilitates the integration of multiple features for a single user, tweet, link, or hashtag, without a priori knowledge on the importance of feature for rumour detection. Finally, the overall p-value is non-parametric, since it does not assume any prior distribution, but integrates any correlation of p-values of different features.

3.5 Global Anomaly Detection

Guided by the requirements for global anomaly detection (R4, R5, R6), we introduce the notion of an anomaly graph (Section 3.5.1), before turning to the computation of the anomalousness of a subgraph (Section 3.5.2), and the detection of a most anomalous subgraph (Section 3.5.3).

3.5.1 Anomaly Graph

Rumour detection using solely local information is not reliable. Local anomalies may be outliers (false positives), as features on social platforms are often noisy [MPLC13] and there are no clear-cut thresholds to filter false positives. Hence, rumour detection shall incorporate information from several elements (entities and relations) of a social graph, each providing a different view on a rumour and, thus, potentially reinforcing each other. A global view is further valuable to differentiate between anomalies that stem from the random nature of social platforms from those that originate from rumours. Finally, the propagation of rumourous information in a social graph helps to understand the rumour structure.

Formally, using the local anomaly detection, each element (entity or relation) in a social graph is associated with a p-value of being rumour-related. Given a social graph $G = (Q, V, E, f)$, this yields an anomaly graph $A = (Q, V, E, p)$, where $p : Q \to [0, 1]$ is a mapping that assign anomaly scores to entities or relations. This anomaly graph is the starting point for the identification of global anomalies, which materialise as subgraphs of the anomaly graph.

3.5.2 Anomalousness of a Subgraph

Rumour structure. Given an anomaly graph $A = (Q, V, E, p)$, a rumour structure is a subgraph of A that is *induced* and *connected*, which are standard graph properties [Die18]. Connectedness is required to cross-check anomaly scores between different elements. The subgraph shall be induced as we shall consider all relations between connected entities as a whole to eliminate false positives.

The anomalousness of a rumor structure is assessed based on:

- *Direct connections*, i.e., the relations (edges) of the graph. While both entities and relations are assigned anomaly scores, we need to conclude on the anomalousness of entities only (e.g., a tweet may be rumourous, while it is not meaningful to consider a tweet-link relation as rumourous). Hence, anomaly scores of a relation and its endpoints need to be unified.
- *Indirect connections* hold between entities that are connected by a path (of length larger than one) in the graph. The longer the path, the smaller the effect of the entities on each other, though.

Anomaly Hypergraph. To incorporate the above aspects, we propose to transform the anomaly graph to an anomaly hypergraph. The idea is to replace every two entities and

the relation between them by a hypernode, which represents the collective information on the entities and the relation, while also providing an aggregated view on their anomaly scores. The hypernode inherits all further relations of the two original entities, i.e., it is connected to all entities to which the original entities had been connected. Formally, given two entities $v_1, v_2 \in V$ and a relation $e = \{v_1, v_2\} \in E$ of an anomaly graph $A = (Q, V, E, p)$, we define the respective hypernode as $v_H = \{v_1, v_2, e\}$ with an anomaly score:

$$p_H(v_H) = \max\{p(v_1), p(v_2), p(e)\} \tag{3.6}$$

Since $p(.)$ is uniformly distributed in $[0, 1]$, $p_H(.)$ also follows a uniform distribution in $[0, 1]$. Here, using *max* for aggregation reduces the chance of false positives, following requirement R4.

Processing all pairs of entities that are connected by a relation in the anomaly graph $A = (Q, V, E, p)$ as detailed above yields an anomaly hypergraph $H = (Q_H, V_H, E_H, p_H)$, with $Q_H \subset [Q]^2$ being a set of modalities, V_H being a set of hypernodes, $E_H \subseteq [V_H]^2$ being a set of edges, and p_H being a mapping function that assigns a anomaly score to each hypernode. Figure 3.3 illustrates this construction.

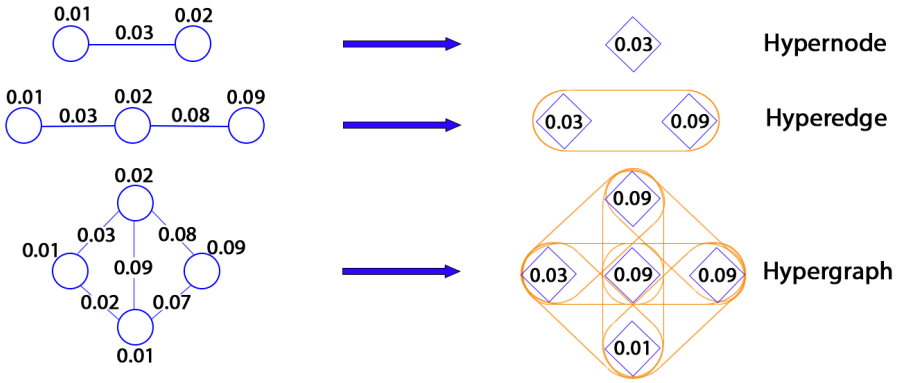


Figure 3.3: Hypergraph construction

Anomalousness measurement. Using the hypergraph H, we strive for a *connected* subgraph S that shows the highest level of anomaly. Since the hypernodes already include the original relations, it is straightforward to revert a subset of connected hypernodes to an induced connected subgraph of the original anomaly graph.

To this end, we first measure the anomalousness of a subgraph, acknowledging the structure of rumours, see requirement R5. We employ the idea of scan statistics [Kul97], which computes the statistical significance of a subgraph S being anomalous without assuming any prior distribution of the subgraph [CN14]:

$$P(S) = \max_{0<\alpha\leq\alpha_{max}} \phi(\alpha, |V_\alpha(S)|, |V(S)|) \tag{3.7}$$

where α_{max} is the maximum statistical significance level ($\alpha_{max} = 0.05$ indicates that the value is *at least* 95% statistical significant), $V(S)$ is the node set of S, $V_\alpha(S) = \{v \in$

$V(S) : p_H(v) \leq \alpha\}$ is the set of nodes in S with anomaly scores that are significant at the confidence level $\alpha > 0$.

To maximize the detection coefficient (see Section 5.2), function $\phi(.)$ shall favour the propagation of rumours, meaning that 'insignificant' nodes ($V(S) \setminus V_\alpha(S)$) are also accepted as long as they are connected with enough 'significant' entities ($V_\alpha(S)$). This is motivated by the dynamic nature of a rumour: Anomaly scores of rumourous entities vary over time and may not be significant at the same time. Moreover, function $\phi(.)$ shall be *non-parametric* (requirement R6), i.e., a function that compares the observed number of α-significant p-values $|V_\alpha(S)|$ to the expected number of α-significant p-values $\mathbb{E}[|V_\alpha(S)|$. Since our p-values are uniformly distributed in $[0, 1]$, we have $\mathbb{E}[|V_\alpha(S)|] = \alpha|V(S)|$. Therefore, we can directly compare $|V(S)|$ and $|V_\alpha(S)|$ as follows [BJ79]:

$$\phi(\alpha, |V_\alpha(S)|, |V(S)|) = |V(S)| \times KL(\frac{|V_\alpha(S)|}{|V(S)|}, \alpha) \tag{3.8}$$

where KL is the Kullback-Leibler divergence defined as $KL(x, y) = x \log(x/y) + (1 - x) \log(\frac{(1-x)}{(1-y)})$. Since $KL(x, y) \geq 0$, it follows that $P(S) \geq 0$ (the higher, the more anomalous). Based thereon, our goal is to detect subgraphs as large as possible (via $|V(S)|$), that have a high confidence level of anomalousness (via $|V_\alpha(S)|/|V(S)|$).

Example 5. *Consider a subgraph S with nodes $V(S) = \{v_1 = 0.02, v_2 = 0.03\}$ and $\alpha_{max} = 0.05$. We have $|V(S)| = 2$. With $\alpha = 0.05$, we have $|V_{0.05}(S)| = 2$ and $\phi(0.05, 2, 2) = 2 \times (1 \log(1/0.05) + 0 \log(0/0.95)) = 2.6$. With $\alpha = 0.02$, we have $\phi(0.02, 1, 2) = 1.1$. With $\alpha = 0.03$, $\phi(0.03, 2, 2) = 3.0$. Therefore, we say that with* at least *95% statistical significance ($\alpha_{max} = 0.05$), we are confident that the anomalousness of $S4$ is $P(S) = \max\{2.6, 1.1, 3.0\} = 3.0$.*

3.5.3 Detection of a Most Anomalous Subgraph

Detecting a rumour structure in an anomaly graph $A = (Q, V, E, p)$ is equivalent to finding a connected subgraph with maximal anomalousness in the anomaly hypergraph $H = (Q_H, V_H, E_H, p_H)$:

$$\arg\max_{S \in \mathcal{S}(H)} P(S) \tag{3.9}$$

where $\mathcal{S}(H)$ contains all possible connected subgraphs of H.

Proposition 1. *Solving Equation 3.9 is NP-hard [CN14].*

Proof (Sketch). With a given α, we can construct a weight function on the node set as $w(v) = 1$ if $p(v) \leq \alpha$ and $w(v) = 0$ otherwise [CN14]. It is known that $\phi(.)$ is monotonically increasing w.r.t. $|V_\alpha(.)|$ [BJ79]. Thus, $\phi(.)$ is monotonically increasing w.r.t. $\sum_{v \in S} w(v)$. Solving Equation 3.9 is now equivalent to finding a solution to the maximum weighted subgraph problem, which is known to be NP-hard [ÁMLM13]. □

As the above problem is computationally expensive, we develop an approximation solution that scales to real-world social graphs. In the context of online social platforms, we argue that such a detection algorithm needs to satisfy two additional requirements:

- *Extensibility.* In practice, multiple rumours may occur at the same time. Hence, we consider a threshold as a relaxation parameter. We then aim at detecting all subgraphs in the anomaly graph that have an anomalousness value above this threshold. Such a threshold may be set based on rumours detected and verified in the past.
- *Incremental processing:* To cope with continuous data generated by social platforms, detection shall be incremental, incorporating new data as it arrives.

An Extensible and Incremental Algorithm. Due to the inherent complexity of Equation 3.9, we present an approach to approximate a solution, see Algorithm 3.1. It takes as input an anomaly graph and a detection threshold, and returns a sorted list of the most anomalous subgraphs that satisfy the threshold. The solution to Equation 3.9 is simply the top-1 in the list. Moreover, in the light of the rumour detection problem (Section 5.2), only the tweet nodes of the output graph may be considered. Since multiple rumours may spread simultaneously on social platforms, however, we include a coverage level K as an input parameter, to cover rumours with smaller anomalousness values.

Algorithm 3.1: Anomalous Subgraphs Detection

input : An anomaly graph $A = (Q, V, E, p)$,
a retain threshold τ (for streaming version),
a coverage level of anomaly K (default = 5),
a specified number of hops Z (default = $log(|V|)$)
output: A sorted list of subgraphs $\mathbb{S}$

1 Construct anomaly hypergraph $H = (Q_H, V_H, E_H, p_H)$ from A;
2 Sort the nodes in H by anomaly score;
3 $\alpha_{max} = 0.05, \mathbb{S} = \mathbb{C} = \emptyset$;
4 **for** $q \in [1, \ldots, |Q_H|]$ **do**
5 **for** $k \in [1, \ldots, K]$ **do**
6 $R = \{v_k\}$, v_k is the k-th most anomalous node in V_H of modality q ;
7 **for** $z \in \{1, ..., Z\}$ **do**
8 $H' = \{v \in V_H \setminus R : \exists v' \in R, \{v, v'\} \in E_H\}$;
9 $\langle S, P(S) \rangle = bestNeighbourhood(H', R, \alpha_{max})$;
10 **if** $S \setminus R \neq \emptyset$ **then** $R = S$;
11 **else break**;
12 $\mathbb{S} = \mathbb{S} \cup \{R\}$;
13 **for** $S \in \mathbb{S}$ **do**
14 **if** $P(S) \geq \tau$ **then** $\mathbb{C} = \mathbb{C} \cup \{S\}$; // `candidate rumours`
15 **return** $\mathbb{S}$;

Our algorithm first expands the subgraphs from a seed node to their neighbours, before greedily optimising the anomaly score for the subgraphs. Specifically, we construct a hypergraph H (line 1), in which each hypernode has an anomaly score, as detailed above. We sort the hypernodes by these scores as this later improves the run-time of the scan statistics subproblem. We then select a root node (line 6), determine its neighbourhood (line 8), and find the subgraph in this neighbourhood with the highest anomaly score (line 9) using Algorithm 3.2 (extended from [Nei12]). The latter greedily retains nodes in the increasing order of p-values (the smaller, the better). Then, we continue to expand the subgraph until our root node set is equal to the most anomalous node set (line 10), i.e., it cannot be expanded further to increase the anomaly score. This guarantees that the subgraph is connected and its anomaly score is maximal [CN14].

Algorithm 3.2: Optimal subgraph in the neighbourhood

input : An anomaly hypergraph H, a root set R, a threshold α_{max}
output: The most anomalous subset S^* and its score $P(S^*)$

```
1  W = {p(v) : v ∈ S} ∪ {α_max};
2  S* = ∅; P(S*) = 0;
3  for α ∈ W do
4      S = ∅; S*_α = ∅; P(S*_α) = 0;
5      for v ∈ sorted(V(H) ∪ R) do
6          S = S ∪ {v};
7          P(S) = φ(α, |V_α(S)|, |V(S)|);
8          if P(S) > P(S*_α) and R ⊆ S then
9              S*_α = S;
10             P(S*_α) = P(S);
11     if P(S*_α) > P(S*) then
12         S* = S*_α;
13         P(S*) = P(S*_α);
14 return ⟨S*, P(S*)⟩ ;
```

Figure 3.4 illustrates the core step of extending the neighbourhood of a root node and finding the optimal subgraph in Algorithm 3.1 (line 6- 10).

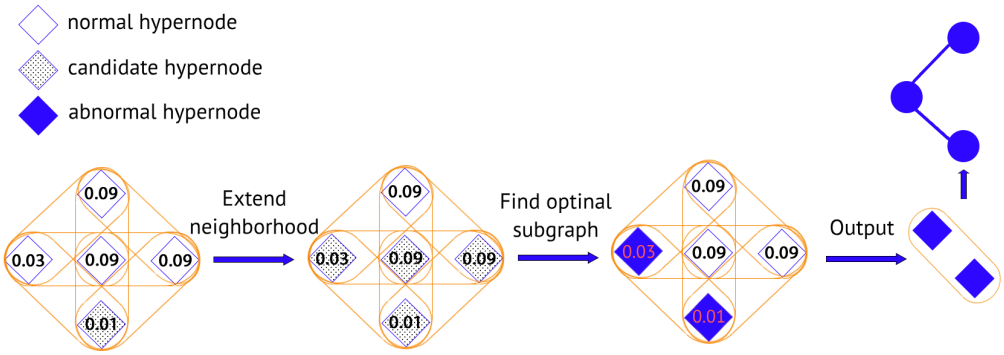


Figure 3.4: Illustration of Algorithm 3.1

Proposition 2. *The output of Algorithm 3.1 is a sorted list of subgraphs in the decreasing order of anomaly level.*

Proof. Algorithm 3.1 processes the nodes in increasing order of p-values (line 6). Since $\phi(.)$ is monotonically increasing w.r.t. $|V_\alpha(.)|$ and monotonically decreasing w.r.t. α and $|V(.)|$ [BJ79], a detected subgraph always has a smaller anomalousness value than its predecessor, which completes the proof. □

3.6 The Streaming Setting

We now lift our approach to a streaming setting. We first discuss how local anomaly scores of a social graph can be computed incrementally (Section 3.6.1), before turning to the incremental computation of anomalous subgraphs detection (Section 3.6.2).

3.6.1 Incremental Anomaly Computation

Recall that computing local anomaly scores is based on historical data. However, in a streaming setting only a window w of data is available, and current observations continuously become historic observations; i.e. $X_{t+|w|} \leftarrow X_t \cup w$. To avoid continuous re-computation of anomaly scores, we propose a heuristic that estimates the score, but works incrementally. Below, we discuss this heuristic for history-based anomaly scores. However, the same approach can also be followed for similarity-based scores.

Intuitively, our approach avoids evaluating Equation 3.1 and Equation 3.2 whenever new data arrives. To this end, we approximate Equation 3.1 with an incremental approach, as long as the respective feature is expected to have no effect on the anomaly score computation. In addition, we discuss how Equation 3.2 can be evaluated efficiently.

Feature-level. To approximate Equation 3.1, we assume that the historical data of a feature of an element x (entity or relation), i.e. $f_{i,j}(X_{T-1}) = \{f_{i,j}(x_t) : x_t \in X_{T-1}\}$ where T is the current timestamp, follows a normal distribution. Note that we consider this assumption solely in the streaming setting, as it yields runtime improvements by not using historic data. In practice, the anomaly scores can be justified by periodic updates from historic data. This distribution, denoted by $N_{j,x}(\mu, \sigma)$, is induced by the empirical mean μ and standard deviation σ computed from historic data. The empirical mean μ and standard deviation σ are updated incrementally as new data arrives:

$$\mu_{t+1} = \frac{\mu_t \times t + x_{t+1}}{t+1}; \quad \mu'_{t+1} = \frac{\mu_t \times t + x^2_{t+1}}{t+1}; \quad \sigma_{t+1} = \sqrt{\mu'_{t+1} - \mu^2_{t+1}}$$

as derived from $\mu = \mathbb{E}[X]$ and $\sigma = \sqrt{\mathbb{E}[X^2] - \mathbb{E}^2[X]}$.

Under the above assumption, Equation 3.1 is approximated using μ and σ. Equation 3.1 essentially counts the number of past values $f_{i,j}(X_{T-1})$ that are greater than the current observation $f_{i,j}(x_T)$. Given a new observation $f_{i,j}(x_{T+1})$ and the historical data captured by $N_{j,x}(\mu, \sigma)$, we derive the percentile of $f_{i,j}(x)$. This percentile is an approximation of how many past observations are greater than the current one. To compute the percentile, we convert $f_{i,j}(x)$ to a z-critical value:

$$z_{i,j}(x) = \frac{f_{i,j}(x) - \mu}{\sigma}$$

Based thereon, the percentile is computed as follows:

$$P(Z \geq z) = \int_z^{+\infty} \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}} dx$$

The percentile value provides us with approximation of the p-value of a specific feature: $p_j(x = z) = P(Z \geq z)$.

The above approximation is used to determine when Equation 3.1 shall be evaluated from scratch. To this end, we exploit that $p_j(x)$ is used to calculate $p_{min}(x_t) = \min_{j=1...q_i} p(f_{i,j}(x_t))$, while $p_{min}(x_{T+1})$ is compared with other $\{p_{min}(x_t)\}_{t=1...T}$ in Equation 3.2. Thus, $p_j(x)$ has an effect on the anomaly score of entity x only if it is smaller than the smallest value $p_{min}(x_t)$. That is, if $\hat{p}_j(x) < \min\{p_{min}(x_t)\}_{t=1...T}$, we do not need to re-evaluate Equation 3.1. We later demonstrate experimentally that this heuristic

helps to reduce the runtime significantly. However, the heuristic requires us to maintain $\min\{p_{min}(x_t)\}_{t=1...T}$, which is done as part of the computation on the entity-level.

Entity-level. When new data arrives, many terms of Equation 3.2 remain unchanged, such as the anomaly score of a feature of an element in the past, $p_{min}(x_t)$. The only term that needs re-computation is the anomaly score of features at the current timestamp, $p_{min}(x_{T+1})$. Therefore, to evaluate Equation 3.2 efficiently, we maintain all values $p_{min}(x_t)$. Given the requirement of maintaining $\min\{p_{min}(x_t)\}_{t=1...T}$, these values are kept in a sorted list. Evaluating Equation 3.2 then becomes counting the number of values stored in the list before $p_{min}(x_{T+1})$.

3.6.2 Incremental Subgraph Detection

To handle streaming data in the computation of anomalous subgraphs, we realise the following idea: Upon the arrival of new data, the anomaly hypergraph will contain new nodes. For these nodes, we identify whether they are rumour-related due to being connected to existing anomalous subgraphs or inducing a new such subgraph. To this end, we associate nodes which belong to an anomalous subgraph with an identifier of the root node used for expansion (nodes may have several such identifiers). This way, upon adding a node, we immediately identify the subgraphs that it may be related to. These subgraphs can be rumour-related ($\mathbb{S}$ in Algorithm 3.1) or potentially-anomalous ($\mathbb{C}$ in Algorithm 3.1), which we distinguish as follows:

In the case that the new node connects to a rumour-related subgraph, the node is assessed based on a property of Algorithm 3.2. Recall that in Algorithm 3.1, we detect anomalous connected subgraphs by expanding subgraphs from root nodes using their neighbours. For each candidate set, we strive for the maximal connected subgraph (Algorithm 3.2). The algorithm relies on a list of nodes, sorted by their p-values. When a new node arrives, we identify the related anomalous subgraphs (if any) and add the new node to the sorted list. If the p-value of the new node is higher than the value of any other node in the subgraph, the new node is rumour-related and added to the subgraph. If a node can be added to several rumour-related subgraphs, the subgraph with the highest anomalousness value is chosen. Otherwise, in the case that the new node connects to a potentially-anomalous subgraph, Algorithm 3.2 is re-run to identify whether the addition of the node yields a new anomalous subgraph.

3.7 Empirical Evaluation

We evaluated our approach with a large real-world dataset obtained from Twitter. Below, we introduce our experimental setting (Section 3.7.1), data collection methodology (Section 3.7.2), and report characteristics of our data (Section 3.7.3). We show that our approach outperforms baseline methods for rumour detection in terms of effectiveness (Section 3.7.4) and explore the design choices of our model (Section 3.7.5). Next, we evaluate the scalability of our methods, including their use in a streaming setting (Section 3.7.6). Finally, we present an illustrative case study (Section 3.7.7).

3.7.1 Experimental Setting

Metrics. We use the following evaluation metrics:

- The detection *coefficient*, first proposed in [SZN13], can be seen as a combination of precision and recall applied to a graph setting. R^* is defined as the set of rumour-

related entities, whereas R is the set of entities labelled by a rumour detection technique. Then, the measure is defined as: $\textit{Coefficient} = |R^* \cap R|/|R^* \cup R|$.

- The *run-time* of processing a set of tweets.
- The *lag time to detection*, which is the time difference the first occurrence of a rumour (i.e., the first rumour-related entity) and its detection (i.e., a first entity is labelled accordingly).

Baselines. State-of-the-art rumour detection [ZAB+18] is not applicable in our context, as it aims at learning a classification model based on a collection of entities that have been labelled with rumours. Such a collection is typically extracted by a pre-processing step that crawls the data related to a particular event, thereby assuming that the extracted elements can be labelled accordingly. As a result, the performance of these approaches strongly depends on the accuracy of such pre-processing. In our work, we progressively detect rumour-related entities by scanning abnormal signals (entities with high anomaly scores) in the social graph.

This fundamental difference in the taken approach is also reflected in the employed evaluation measures. Existing rumour detection techniques are evaluated using machine learning metrics, applied per rumour. This is not possible for our approach, so that we rely on the detection coefficient, applied per graph entity. In a broad sense, most rumour detection techniques focus on maximizing accuracy, instead of striving for a balance of accuracy and completeness.

Against this background, we consider several baseline methods. We implemented these methods based on the respective papers.

- *Decision* [CMP11]: A decision tree classifier that is based on the Twitter information credibility model. The decision tree is constructed based on several hand-crafted features.
- *Nonlinear* [YLYY12]: An SVM-based approach that uses a set of hand-crafted features, selected for the tweets to classify.
- *Rank* [ZRM15]: A rank-based classifier that aims to identify rumours based on enquiry tweets.

In addition, we also compare our approach with methods based on homogeneous graphs that contain only a single modality. For instance, a tweet graph contains only tweets, while edges between tweets represent that tweets stem from the same user, have retweet relations, or share a keyword. We constructed four such homogeneous graphs, for users, tweets, links, and hashtags, respectively.

Parameters. We set the statistical significance level $\alpha_{max} = 0.05$ (i.e. the result is guaranteed to be *at least* 95% confidence). The coverage level K in Algorithm 3.1 has been varied, so that we can detect multiple rumours at the same time.

For the static version of our approach, our rumour detection algorithm is executed multiple times by gradually extending the historical data $X_t = \{x_1, \ldots, x_t\}$ from the first day ($t = 1$) to the last day of each dataset. At each extension, all tweets in detected rumours will be removed to avoid that some rumours in the future will have smaller anomaly scores than the past (and thus the p-values might not be high enough with 95% confidence threshold).

For the incremental version, we set the window size $|w|$ to 12 hours; i.e. the historical data is defined by $X_t = \{x_{t-|w|}, \ldots, x_t\}$. Again, all tweets in detected rumours are

removed. Note that, however, we cannot remove other types of entities (users, hashtags) since they potentially participate in different rumours. The threshold τ to retain the candidate rumours is set by the 20-quantiles of the anomalousness values of returned subgraphs.

Experimental environment. All results have been obtained on an Intel Core i7 system (2.8 Ghz, 32GB RAM).

3.7.2 Data Collection

Rumour collection. Snopes is a world-leading rumour-debunking service. Unlike other organizations such as Politifact and Urbanlegends, it is considered to be objective when evaluating the veracity of rumours [Net]. Snopes editors investigate each rumour along different dimensions and provide an argumentative report as shown in Table 3.2. For example, the claim describes the rumour succinctly and the rating represents its truth value according to the fact-checker.

Table 3.2: Information about a rumour.

Attribute	**Example**
id	trump-aid-puerto-rico
date	10/2/2017
genesis tweet	[..] President Trump has dispatched 140 helicopters [..]
sources of veracity	press reports, local officials, organizations
rating	MIXTURE [Snob]

Multi-model social graph construction. Twitter is a large social platform with tweets covering various domains such as politics and crime. It is frequently used by users to express their opinions in a timely manner, e.g., by retweeting others, which provides insights into how rumours propagate. These characteristics make Twitter data particularly suitable for evaluating rumour detection methods.

We followed the dataset construction process described in [KCJ17]. For each rumour, we identify its fingerprint, which is a set of keywords. Then, we use these keywords to search for tweets that are related to this rumour using Spinn3r [Spi]. We take the ID of a Snopes article as the starting point to create the fingerprint of a rumour. If the ID is not unique or too general, keywords are manually selected from the rumour's claim and the respective Snopes article. Applying modifications to these keywords provided us with a set of search queries to identify rumourous tweets. Since the queries may not identify all tweets that are rumour-related, we also considered retweets. To obtain negative samples, we collected further tweets from the timelines of users that authored rumourous tweets and of other users identified by retweets of regular tweets.

At this point, the social graph contains two entity types (tweets and users) and one relation type (user-tweet). The remaining entity and relation types are constructed as follows. For each tweet, we extract the links using regular expressions and crawl the corresponding articles, which results in a tweet-link relation. The link-hashtag relation is created by connecting an article to any hashtag it mentions. The user-hashtag relation is created by connecting a user to a hashtag they used in their tweets. The user-link relation stems from connections of a user to an article they mentioned in their tweets.

Feature engineering. Features of each individual entity are engineered as follows. Static features (similarity-based) have been extracted directly from the Twitter REST

API [KLPM10], including user features such as registration age. To assess the credibility of a user, we relied on Tweetcred framework [GKCM14], which is an aggregation of 45 characteristics such as #retweets, #favorites, #replies, and presence of swear words into Likert Scale (score 1-5). The credibility feature of linked articles was assessed using the Alexa ranking (higher ranking, higher credibility). Popularity of hashtags was quantified using semantic ranking [BJV15]. The linguistic style of tweets and linked articles was evaluated using OpenIE framework [MWDNM14b]. Each linguistic feature is measured as the fraction of English words in a tweet that reflect the writing style of the user. Six linguistic features are used: discrepancy words (e.g., could, would), tentative words (e.g., perhaps), filter words (e.g., I mean), punctuations, swear words (e.g., damn), and exclusion words (e.g., but).

Dynamic features (history-based) are extracted using the Twitter Streaming API [MPLC13]. For instance, the number of retweets of a tweet is collected over time by monitoring the respective tweet. Similarly process is used for status frequencies, numbers of followers and friends of a user. Numbers of tweets as well as mentions of hashtags and links were obtained using this way.

Similarly, data is collected for features of relations. For example, the difference between the time mentioned in a tweet and given in a linked article is assessed for all tweets in a specific time window. Then, upon receiving a tweet that links to an article, the respective time difference can be compared to those observed for historic data. Location and event features, in turn, are binary and capture whether the tweet and link originate from the same location or event.

Datasets. The collected data comprises 4 million tweets, 3 million users, 28893 hashtags, and 305115 linked articles, revolving around 1022 rumours from 01/05/2017 to 01/11/2017. This period was chosen as it contains several rumours, e.g., related to the Las Vegas shooting and information published by the US administration. Our data spans over 20 different domains, available at [Snod]. Here, we report results for the most popular ones:

- *Politics*: rumours related to all political issues.
- *Fraud & Scam*: rumours related to online hoax/scam entreating users to share posts and photographs under the false premise of a greater good.
- *Fauxtography:* rumours related to images or videos circulating on the Web.
- *Crime*: rumours related to criminology and incidents, such as the Las Vegas shooting.
- *Science & Technology:* rumours related to scientific myths and exaggerated technological inventions.

Each of the datasets is a full view of the social graph. The modelled entity types, relation types, and features are summarised in Table 3.1.

3.7.3 Understanding Rumour Characteristics

What are the rumours about? In order to understand the diffusion of rumours on social platforms, we plot the distribution of rumours with their respective tweets in Figure 3.5. The top-3 domains with the most number of rumours and tweets are *Politics*, *Fraud & Scam*, and *Fauxtography*. In total, they comprise over 80% of number

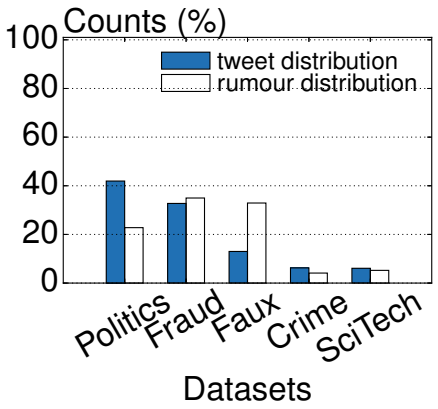


Figure 3.5: Data distributions

of rumours and tweets. This implies that rumours are easily spread in the domains where being right or wrong is rather subjective.

We also observe discrepancies between the number of rumours and the number of tweets in each domain. Although the majority of tweets is in the Politics domain, the number of rumours belonging to this domain is only the third highest. As political rumours are controversial, they tend to attract more interactions, leading to a high number of tweets [VRA18]. On the other hand, although more than 30% of rumours are Fauxtography, only 10% of the tweets belong to this category. An explanation may be that false pictures are easy to create, but may not deceive people easily.

Who post rumours? To investigate the features of rumour-related users, Figure 3.6 displays boxplots of the relations between the number of friends, followers, lists [KJMO10] (groups on Twitter that a user can subscribe to) and likes of a user and the domain of rumours to which they contributed. Interestingly, users who post fraud & scam tweets have lower numbers of features on average in comparison with other domains. Moreover, there seems to be no correlation between the number of friends and followers and the domain of rumours.

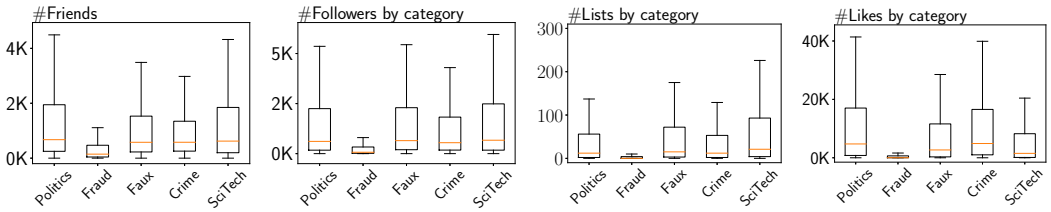


Figure 3.6: Relations between user features and rumours

Where are the rumours from? Figure 3.7 shows the number of users who tweet about rumours by country. Here, the most prominent countries are English-speaking (US, UK) or populous (China, India). The majority of users in our dataset, however, resides in the US, with nearly 0.4M users. Figure 3.8 analyses whether there is an indication that the location of the users affects the domain of their tweets. The top popular domains for most countries are *Politics*, *Fraud*, and *Faux*, which is similar to the top domains in overall. This fits with the data collection period after the 2016 US presidential election.

In Figure 3.9, we show a histogram of the numbers of users who post tweets related to different rumours. The histogram follows a long-tail distribution in which most users

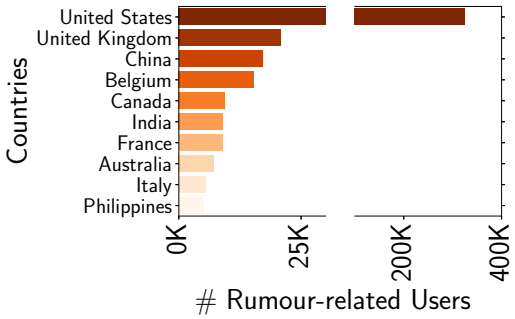


Figure 3.7: Users by countries

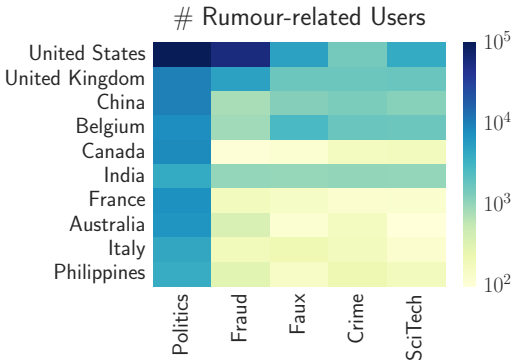


Figure 3.8: Users by domains

tweet about 1-2 rumours. There are users who tweet about more than 100 rumours. However, their number is extremely small. Analysing these users, we identify several interesting characteristics. The accounts who post about most rumours are extremely similar. We suspect that they are bots or part of a network. Given our focus on rumour detection, however, we refer to [VRA18] for an in-depth analysis of user accounts.

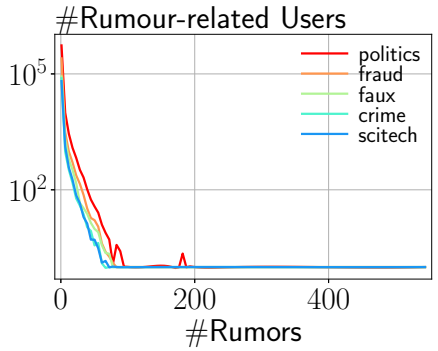


Figure 3.9: Users who tweet-/retweet rumours

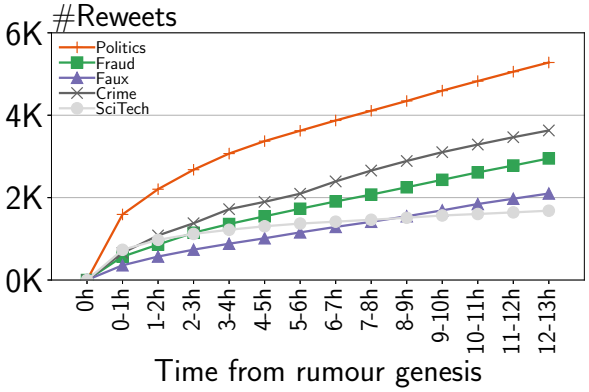


Figure 3.10: Propagation of rumours

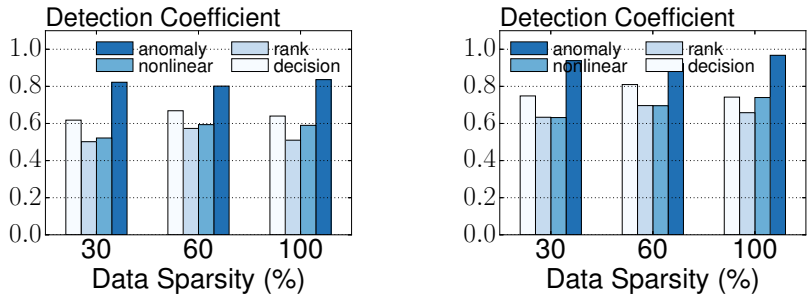
How do rumours propagate? To illustrate the propagation of rumours, we collect the number of retweets per tweet, which is a measure of its influence. Figure 3.10 shows the number of retweets per rumor per domain in the first 13 hours.

We observe that political rumours are extremely bursty. In the first hour, the average number of retweets of these rumours is over 1000, which indicates that these rumours can spread in a short amount of time. After the first hour, these rumours keep propagating extremely fast, following a linear trend. Therefore, it is important that rumours belonging to this domain are detected early. On the other hand, rumours in other domains follow a log-scale increase after the first hour. In addition, rumours in these domains are not as bursty. The number of retweets after the first hour is moderate as most of them have less than 500 retweets in the first hour.

3.7.4 Effectiveness of Rumour Detection

Detecting rumourous tweets. We evaluate the detection coefficient of our approach versus the baseline methods in Figure 3.11 for the domains *Politics* and *Crime* (the same trends emerge for the other domains). We vary the amount of rumours contained in the

dataset, i.e. data sparsity, by randomly removing some rumours, so that the remaining rumours cover 30%, 60%, 100% of the original count.



(a) Politics (b) Crime

Figure 3.11: Rumour Detection Coefficient across datasets

In general, our approach outperforms the baseline methods in the detection of rumour-related tweets. For instance, taking the results of the *Politics* dataset, when considering 30% of the rumours, our approach achieves a coefficient of 0.82, whereas the best baseline method achieves solely a coefficient of 0.62.

Going beyond the detection of tweets. Our multi-modal approach enables not only the detection of rumour-related tweets, but also rumour-related users, hashtags and links. We therefore evaluated the effectiveness of rumour detection for these modalities, in comparison with the baseline methods. As the baseline methods detect solely rumour-related tweets,we used these tweets to determine rumour-related users, hashtags, and links that are their direct neighbours in the social graph. We assessed the performance of our approach and the baseline methods in terms of the achieved coefficient, when varying the amount of rumours contained in the dataset.

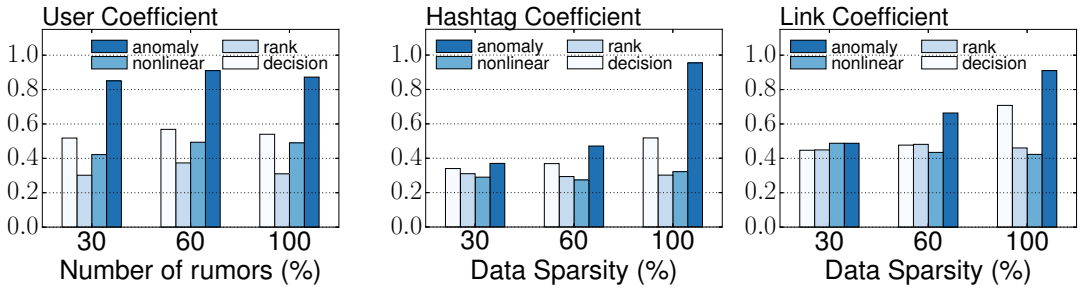


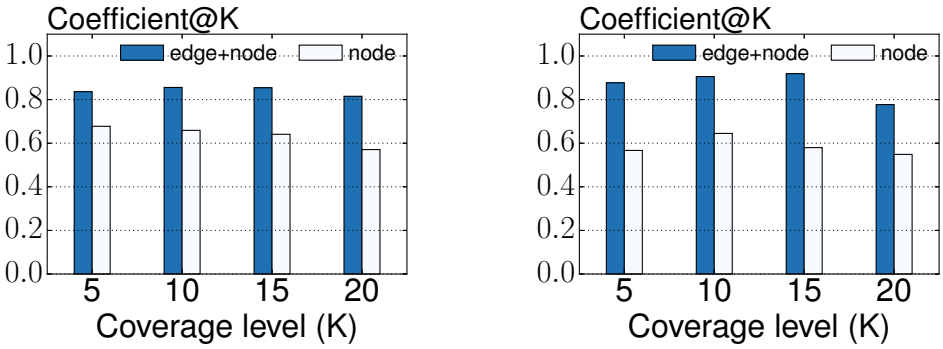
Figure 3.12: Coefficients for different modalities

Figure 3.12 shows the results obtained for users, links and hashtags on *Politics* (results for other datasets are similar). Our approach still outperforms in the detection of rumour-related users, links, and hashtags. This is expected as our approach incorporates multiple modalities explicitly, which yields a synergistic effect when trying to detect rumour-related entities of different types.

3.7.5 Model Design Choices

Effects of Relations. We analyse the effect of considering relations of the social graph when detecting rumours. To this end, we detected anomalies using only entities (*node*)

and compare the results to our actual approach (*edge+node*). We varied the coverage level in Algorithm 3.1 to obtain multiple anomalous subgraphs.

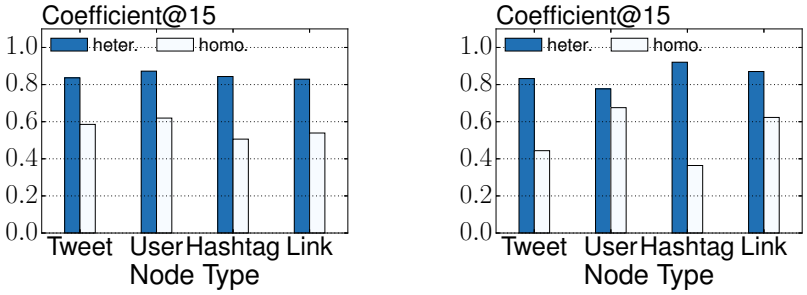


(a) Politics (b) Crime

Figure 3.13: With vs. without relations

The results in Figure 3.13 show that using solely entities yields worse coefficients, e.g., a value of 0.64 instead of 0.85, when considering $K = 15$ in the *Politics* dataset (again, trends are consistent over all domains). This highlights that relations constitute an important source of information for rumour detection.

Effects of Multi-Modality. We further evaluated the impact of multi-modal information, by comparing our approach with rumour detection based on homogeneous graphs, built of a single modality. The respective modality is then taken as the target for rumour detection, e.g., the user graph is used to detect rumour-related users. We measure the detection coefficient, while considering the best coverage level $K = 15$ from the previous experiment.



(a) Politics (b) Crime

Figure 3.14: Heterogeneous vs. homogeneous graphs

As illustrated in Figure 3.14 for two domains, the multi-modal social graph yields a better coefficients. This underlines the importance of a rich model, with multiple modalities, for rumour detection.

3.7.6 Scalability and Streaming Settings

Effects of data size. This experiment compared the non-incremental and incremental versions of our approach. We constructed sub-datasets to vary the number of nodes in the social graph of the *Politics* dataset from 10^3 to 10^6 and compare the observed coefficient and run-time. Figure 3.15 shows that the incremental computation indeed

improves the run-time of our approach, halving the time needed to process a graph of size 10^6. Moreover, the error introduced by incremental computation stays within reasonable bounds.

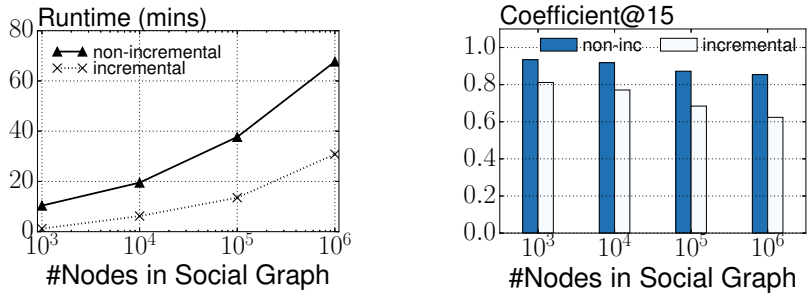
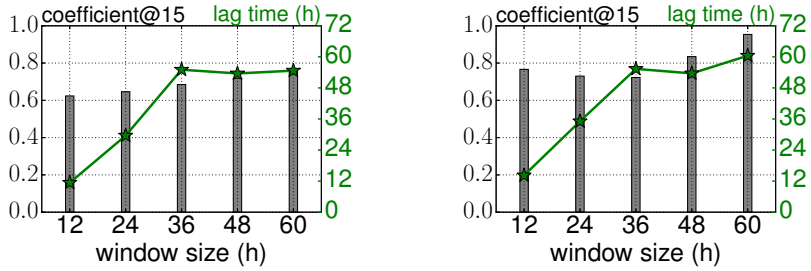


Figure 3.15: Incremental vs. non-incremental

Effects of window size. We varied the window size, from 12 to 60 hours, while considering the coverage level $K = 15$. The results in terms of coefficients and lag time to detection are shown in Figure 3.16. With larger windows, the coefficient increases, since rumour detection exploits more information. The lag time to detection also increases, until reaching a plateau. Again, this is due to the amount of available information. Initially, some rumours cannot be detected and thus do not affect the lag time. With larger windows, these rumours are detected and increase the lag time.

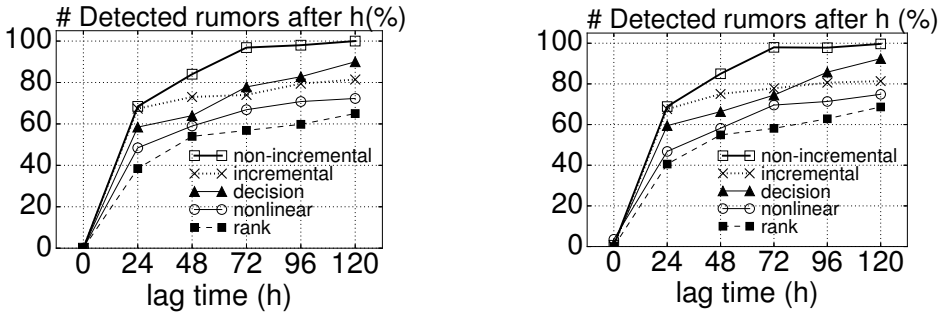


(a) Politics (b) Crime

Figure 3.16: Streaming setting: effects of window size

Distribution of lag time. Further, we studied the relation between lag time and detection accuracy. For our incremental approach, we computed the lag time for each rumour and aggregate them into several bins. For all other methods, we constructed datasets with varying detection deadlines θ, controlling that for each rumour, only tweets from the start of the rumour (θ_0) until $\theta_0 + \theta$ are kept. We then report the percentage of detected rumours for each such deadline. According to Figure 3.17, our approaches outperform the baseline methods, especially for small lag times. For instance, in the *Politics* dataset, with a lag time of 48 hours, our non-incremental approach detects 84% of rumours, whereas the best baseline achieves 64%.

Average delay analysis. We provide a fine-grained view of the lag time by computing the difference between the timestamps at which the same rumour was *first* detected (i.e. any tweet related to that rumour is flagged) by different methods. Table 3.3 presents the analysis within 1 day after the genesis of rumours. Our approach detect rumours earlier than the baselines a few hours in average.



(a) Politics (b) Crime

Figure 3.17: Timeliness of rumour detection

Baseline	Rumours detected	Average delay
our	68.29%	+0.0h
decision	59.46%	+1.7h
nonlinear	47.73%	+2.3h
rank	38.23%	+ 3.1h

Table 3.3: Delay analysis (within 1 day)

3.7.7 Case Study

Effects of timeline. Figure 3.18 highlights some detected rumours from the *Crime* dataset along a timeline. Most of the rumours are related to the Las Vegas shooting, one of the biggest events of the year that attracts many hoaxes, fake news, and viral misinformation [Snoc]. It plots the hourly numbers of tweets for each rumour. Here, most rumours occurred around on October 2, the date of the incident. Most rumour-related tweets are about the shooting being caused by a member of ISIS. Also, two days after the incident, there was a rumour that the shooter had an accomplice, which create another peak in the number of tweets (second-shooter rumour).

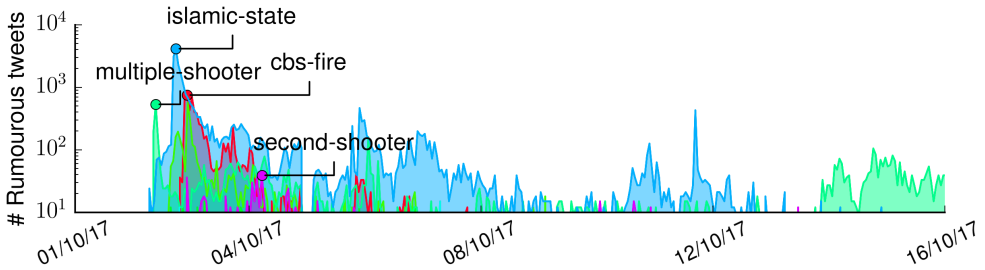
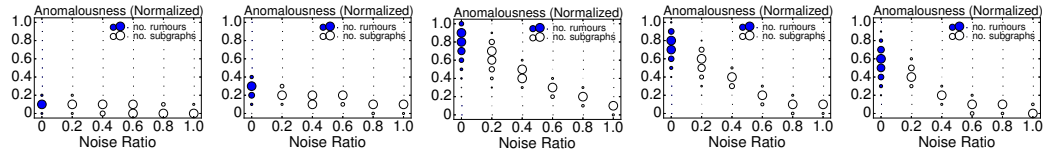


Figure 3.18: Timeline of rumours about the Las Vegas shooting in October 2017

Correctness of anomaly scores. Figure 3.19 depicts the correctness of our anomalousness measure on subgraphs. When a rumour happens (genesis), we compute its anomalousness score, do the same at $\pm$ 1 day and $\pm$ 2 days, and then normalize by the maximum values among all rumours. These scores are compared with those of other subgraphs, which are constructed by randomly adding regular tweets into the rumours (this noise ratio is varied from 0.0 to 1.0). Finally, we do a histogram by counting the number of rumours and subgraphs with scores that fall into 0.1-bins.



(a) 2 days before genesis (b) 1 day before genesis (c) Genesis time (d) 1 day after genesis (e) 2 days after genesis

Figure 3.19: Correctness of anomaly scores

At the genesis, the scores of positive samples (i.e. rumours) turn out to be significantly higher than noisy samples (i.e. other subgraphs), supporting effective detection. Before the genesis, anomaly scores are small and nearly uniform, as historic data is not anomalous. After the genesis, the scores decrease. Yet, they are still relatively high, since anomalies are still present, which captures the temporal movement of rumours.

3.8 Summary

This chapter proposed an approach for rumour detection that is grounded in the anomalies of a social graph. Unlike traditional approaches that focus only on accuracy, we optimised the detection coefficient, which represents the trade-off between accuracy and completeness. We presented a two-step detection approach that detects anomalies at the local and global level. While the former increases the completeness of detection by reducing false negatives, the latter optimises the detection accuracy by reducing false positives. Our experiments showed that our method is effective and efficient, detecting rumours early and accurately. It outperformed several baselines in both static and streaming settings.

Chapter 4

Misinformation Validation: The Case of Minimal-Effort Fact-Checking

User Guidance for Efficient Fact Checking

VLDB 2019

Maximal Fusion of Facts on the Web with Credibility Guarantee

Information Fusion Journal 2019

The Web constitutes a valuable source of information. In recent years, it fostered the construction of large-scale knowledge bases, such as Freebase, YAGO, and DBpedia. The open nature of the Web, with content potentially being generated by everyone, however, leads to inaccuracies and misinformation. Construction and maintenance of a knowledge base thus has to rely on fact checking, an assessment of the credibility of facts. Due to an inherent lack of ground truth information, such fact checking cannot be done in a purely automated manner, but requires human involvement.

In this chapter, we propose a comprehensive framework to guide users in the validation of facts, striving for a minimisation of the invested effort. Our framework is grounded in a novel probabilistic model that combines user input with automated credibility inference. Based thereon, we show how to guide users in fact checking by identifying the facts for which validation is most beneficial. Moreover, our framework includes techniques to reduce the manual effort invested in fact checking by determining when to stop the validation and by supporting efficient batching strategies. We further show how to handle fact checking in a streaming setting. Our experiments with three real-world datasets demonstrate the efficiency and effectiveness of our framework: A knowledge base of high quality, with a precision of above 90%, is constructed with only a half of the validation effort required by baseline techniques.

4.1 Introduction

Extracting factual knowledge from Web data plays an important role in various applications. For example, knowledge bases such as Freebase [Fre], YAGO [YAG] and DBpedia

rely on Wikipedia to extract entities and their relations. These knowledge bases store millions of facts, about society in general as well as specific domains such as politics and medicine. Independent of the adopted format to store facts, extraction of factual knowledge first yields candidate facts (aka claims), for which the credibility needs to be assessed. Given the open nature of the Web, where content is potentially generated by everyone, extraction of claims faces inaccuracies and misinformation. Hence, building a knowledge base from Web sources does not only require conflict resolution and data cleansing [DSS12], but calls for methods to ensure the credibility of the extracted claims, especially in sensitive domains, such as healthcare [MWDNM14b].

To assess the credibility of claims, automated methods rely on classification [LGMN12] or sensitivity analysis [WAL+14]. While these methods scale to the volume of Web data, they are hampered by the inherent ambiguity of natural language, deliberate deception, and domain-specific semantics. Consider the claims of 'the world population being 7.5 billion' or 'antibiotics killing bacteria'. Both represent common-sense facts. Yet, these facts have been derived from complex statistical and survey methods and, therefore, cannot easily be inferred from other basic facts.

When relying on accurate facts, incorporating manual feedback is the only way to overcome the limitations of automated fact checking. However, eliciting user input is challenging. User input is expensive (in terms of time and cost), so that a validation of all claims is infeasible, even if one relies on a large number of users (e.g., by crowdsourcing) and ignores the overhead to resolve disagreement among them. Also, claims are not independent, but connected in a network of Web sources. An assessment of their credibility thus requires effective propagation of user input between correlated claims. Finally, there is a trade-off between the precision of a knowledge base (the ratio of credible facts) and the amount of user input: The more claims are checked manually, the higher the precision. However, user input is commonly limited by some budget.

Our approach. This chapter presents a comprehensive process for guiding users in fact checking, adopting a pay-as-you-go approach. We present a novel probabilistic model that enables us to reason on the credibility of facts, while new user input is continuously incorporated. By (i) inferring the credibility of non-validated facts from those that have been validated, and by (ii) guiding a user in the validation process, we reduce the amount of manual effort needed to achieve a specific level of result precision. Credibility inference and user guidance are interrelated. Inference exploits mutual reinforcing relations between Web sources and claims, which are further justified based on user input. Moreover, a user is guided based on the potential effect of the validation of a claim for credibility inference.

Efficient user guidance further requires to decide: (i) when to terminate validation to avoid wasting resources on marginal improvements of the quality of the knowledge base; (ii) how to group claims for batch processing to reduce the impact of set-up costs in validation (a user familiarising with a particular domain); and (iii) how to handle continuous arrival of new data to avoid redundant computation. Our novel model enables us to address these aspects.

Our contributions are summarised as follows:

- *Approach to Guided Fact Checking:* Section 4.2 formalises the setting of fact checking and, based thereon, formulates the problem of effort minimisation. We further introduce an iterative approach to guide a user in the validation process and highlight requirements for its instantiation.
- *Probabilistic credibility inference:* Section 4.3 addresses the need for a method to reason on the credibility of facts. We introduce a probabilistic model for fact check-

ing, based on Conditional Random Fields, and show how to perform incremental inference based on user input. Aiming at pay-as-you-go validation, we show how to derive a trusted set of facts based on our model.
- *Probabilistic user guidance:* Section 4.4 presents strategies to guide users, i.e., to select the claims for which validation is most beneficial. These strategies target the reduction of uncertainty in our probabilistic model for fact checking.
- *Complete validation process:* Section 4.5 combines our mechanisms for credibility inference and user guidance to obtain a comprehensive validation process. We also show how to achieve robustness against erroneous user input.
- *Methods for effort reduction:* Section 4.6 introduces techniques for early termination of the validation process and batch selection. The former is based on signals that indicate convergence of our probabilistic model and, thus, of the quality of the derived knowledge base. The latter selects groups of claims for validation based on the benefit of their joint validation. Since this selection problem turns out to be intractable in practice, we propose a greedy top-k algorithm, which comes with performance guarantees.
- *Streaming fact checking:* Section 4.7 shows how to handle continuously arriving data by an adaptation of our validation process that features stochastic approximation and reuse of model parameters.

We evaluate our techniques with three large-scale datasets (Section 4.8) of real-world claims. We demonstrate low response times for claim selection (<0.5s) and high effectiveness of guiding users in their validation efforts. To obtain a knowledge base of high quality (>90% precision), only a half of the effort of baseline techniques is required. Finally, we review related work (**??**) and conclude (Section 5.6).

4.2 Guided Fact Checking

4.2.1 Setting

We model the setting of fact checking by means of a set of data sources $\mathcal{S} = \{s_1, \ldots, s_u\}$, a set of documents $\mathcal{D} = \{d_1, \ldots, d_m\}$, and a set of candidate facts, or short claims, $\mathcal{C} = \{c_1, \ldots, c_n\}$. A source could be a user, a website, a news provider, or a business entity. It provides multiple documents, each often being textual (e.g., a tweet, a news item, or a forum posting) and involving a few claims. The representation of a claim (e.g., unstructured text or an RDF triple) is orthogonal to our model. However, a claim can be referenced in multiple documents, it depends on a specific process for information extraction how the link between claims and documents is established (see Section 4.8.1).

A claim $c \in \mathcal{C}$ represents a binary random variable, where $c = 1$ and $c = 0$ denote that the claim is credible or non-credible, respectively. In fact checking, however, these values are not known, so that we consider a probabilistic model P, where $P(c = 1)$, or $P(c)$ for short, denotes the probability that claim c is credible. Combining the above notions, the setting of fact checking is a tuple $Q = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P \rangle$, also referred to as a *probabilistic fact database*.

A knowledge base is constructed from such a database by deriving a trusted set of facts. We formalise this construction by a grounding function $g : \mathcal{C} \to \{0, 1\}$, labelling claims as credible ($g(c) = 1$) or non-credible ($g(c) = 0$).

In fact checking, claims are validated manually by a user, which is represented by a binary model of user input. A claim c is either confirmed as credible, which yields $P(c) = 1$, or labelled as non-credible, so that $P(c) = 0$.

As an example, consider the *Snopes* dataset [dat17b], a collection of 4856 claims derived from 80421 documents of 23260 sources, such as news websites, social media, e-mails, etc. For instance, this dataset comprises the claim that *eating turkey makes people especially drowsy*. This claim can be found in documents of various Web sources, among them earthsky.org [ec19a], webmd.com [ec19c], and kidshealth.org [ec19b]. In the Snopes dataset, claims have been validated by expert editors, which corresponds to the user input in our model. It labels the aforementioned example claim as non-credible [eca19].

4.2.2 Effort Minimisation

Adopting the above model, the grounding g to derive a trusted set of facts is partially derived from user input. However, manual validation of claims is expensive, in terms of user hiring cost and time. User input is commonly limited by an effort budget, which leads to a trade-off between validation accuracy and invested effort.

Going beyond this trade-off, we aim at minimising the user effort invested to reach a given validation goal. We consider fact checking as an iterative process with a user validating the credibility of a single claim in each iteration. This process halts either when reaching a validation goal or upon consumption of the available effort budget. The former relates to the desired result quality, e.g., a threshold on the estimated credibility of the grounding. The latter defines an upper bound for the number of validations by a user and, thus, iterations of the validation process.

Formally, given a probabilistic fact database $\langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P \rangle$, fact checking induces a *validation sequence*, a sequence of groundings $\langle g_0, g_2, \ldots, g_n \rangle$ obtained after incorporating user input as part of n iterations of a validation process (i.e., any g_i is a prediction of the model). Given an effort budget b and a validation goal Δ, a sequence $\langle g_0, g_1, \ldots, g_n \rangle$ is *valid*, if $n \leq b$ and g_n satisfies Δ. Let $\mathcal{R}(\Delta, b)$ denote a finite set of valid validation sequences that can be created by instantiations of the validation process. Then, a validation sequence $\langle g_0, g_1, \ldots, g_n \rangle \in \mathcal{R}(\Delta, b)$ as *minimal*, if $n \leq m$ for any validation sequence $\langle g'_0, g'_1, \ldots, g'_m \rangle \in \mathcal{R}(\Delta, b)$.

Problem 2 (Effort Minimisation)**.** *Let $\langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P \rangle$ be a probabilistic fact database and $\mathcal{R}(\Delta, b)$ a set of valid validation sequences for an effort budget b and a goal Δ. The problem of* effort minimisation in fact checking *is the identification of a minimal sequence $\langle g_0, g_1, \ldots, g_n \rangle \in \mathcal{R}(\Delta, b)$.*

The validation goal could be the precision of the final grounding g_n, estimated by cross validation. Note that, in theory, Problem 2 could have no solution—the effort budget may be too small or the validation goal may be unreachable. However, for practical reasons, there needs to be a guarantee that the validation process terminates.

Solving Problem 2 is challenging, mainly for two reasons. First, claims are not independent, but subject to mutual reinforcing relations with Web sources and documents. Consequently, the validation of one claim may affect the probabilistic credibility assessment of other facts. Second, the problem is computationally hard: Finding an optimal solution quickly becomes intractable, since all permutations of all subsets (of size $\leq b$) of claims would have to be explored.

4.2.3 Outline of the Validation Process

To address the problem of effort minimisation, we argue that a user shall be guided in the validation of claims. In essence, user input shall be sought solely on the 'most

promising' unverified facts, i.e., those for which manual validation is expected to have the largest impact on the estimated credibility of the resulting grounding.

Let $\langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P \rangle$ be a probabilistic fact database. Our validation process continuously updates the grounding g to validate claims in a pay-as-you-go manner, by:
(1) *selecting* a claim c for which feedback shall be sought;
(2) *eliciting* user input on the credibility of c, which either confirms it as credible or labels it as non-credible;
(3) *inferring* the implications of user input on the probabilistic credibility model P;
(4) *deciding* on the grounding g that captures the facts that are assumed to be credible.
In the above process, steps (1), (3), and (4) need to be instantiated with specific methods. An example for a straight-forward instantiation would be a validation process that:

- *selects* a claim c randomly for validation;
- limits the *inference* to claim c, setting either $P(c) = 1$ or $P(c) = 0$, not changing $P(c')$ for any claim $c' \neq c$;
- *decides* that a claim c is credible, $g(c) = 1$, if and only if it holds $P(c) \geq 0.5$.

In the remainder, we present methods for a more elaborated instantiation of the above process. We introduce a probabilistic model for fact checking that captures the mutual reinforcing relations between Web sources and claims. This enables us to *infer* the implications of user input beyond the claims that have been validated, and based thereon, *decide* on the grounding while incorporating the relations between sources and claims. Also, the model enables conclusions on the claims that shall be *selected*. Unverified claims for which validation is most beneficial for the inference will be chosen. Our model further helps to identify suspicious user input, i.e., claims that may have been validated by mistake.

We then address aspects of practical relevance, which are not captured in Problem 2. Validation may converge *before* the validation goal is reached and the effort budget has been spent. If so, further user input leads to diminishing improvements of the quality of the grounding and the validation process may be terminated. We show how our model enables the detection of such scenarios by decision-support heuristics.

In practice, users that validate claims face significant set-up costs, implied by the need to familiarise with claims of a particular domain. It therefore increases user convenience and efficiency if the validation process considers a batch of claims per iteration. We support such batching by a greedy top-k strategy to select a set of claims with a high *joint* benefit for credibility inference.

Moreover, in many applications, new sources, documents, and claims arrive continuously. We thus illustrate how the above process can be lifted to a streaming setting by exploiting online algorithms for inference and reusing parameters of our underlying model.

4.3 Credibility Inference

This section presents a probabilistic model for fact checking (Section 4.3.1), before turning to mechanisms for incremental inference (Section 4.3.2) and the instantiation of a grounding (Section 4.3.3).

4.3.1 A Probabilistic Model for Fact Checking

Sources of uncertainty. Claims are assessed by means of documents from Web sources. These documents are encoded using a set of features. We abstract from the specific

nature of these features, but take into account that the trustworthiness of a source and the language quality of a document have a strong influence on the credibility of the claims. We capture these features as follows. A source $s \in \mathcal{S}$ is associated with a feature vector $\langle f_1^{\mathcal{S}}(s), \ldots, f_{m_{\mathcal{S}}}^{\mathcal{S}}(s)\rangle$ of $m_{\mathcal{S}}$ source features. In the same vein, $\langle f_1^{\mathcal{D}}(d), \ldots, f_{m_{\mathcal{D}}}^{\mathcal{D}}(d)\rangle$ is a vector of $m_{\mathcal{D}}$ document features, assigned to each document $d \in \mathcal{D}$.

Features of sources and documents interact with each other, and with the credibility of claims. A claim's credibility depends on both, the trustworthiness of the source and the language quality of the document, which we call a *direct relation*. A claim is more likely to be credible, if it is posted by a trustworthy source using objective language. Yet, the intentions of a source, and thus its trustworthiness, may change over different contexts and hence documents. Therefore, we also reason about the credibility of claims via an *indirect relation*, exploiting that documents of different sources may refer to the same claim. For example, a source disagreeing with a considered credible by several sources shall be regarded as not trustworthy.

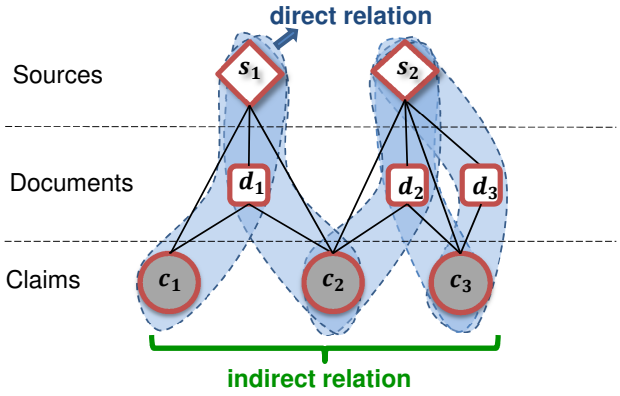


Figure 4.1: Relations in a probabilistic fact database.

The Conditional Random Field model. To model these relations, and eventually derive the assignment of credibility probabilities, we rely on a Conditional Random Field (CRF) [Elk08], see Figure 4.1. We construct a CRF as an undirected graph of three sets of random variables, $S, D, \mathcal{C}$ for sources, documents, and claims. Here, S and D are sets of real-valued variables that represent trustworthiness of sources and language quality of documents, respectively, based on the aforementioned features. Set $\mathcal{C}$ is the set of binary variables introduced in Section 4.2.1, each variable representing a claim's credibility. Direct relations are captured by relation factors in the CRF, also called cliques since they always involve three random variables (source, document, claim). Any random variable can be part in multiple cliques, reflecting the indirect relations. This implies a factorization of cliques to compute the joint probability distribution.

In this model, S and D are observed variables. As an output variable, we consider a categorical variable C that represents credibility configurations of claims. A possible value o of C, called configuration, is an assignment $o : \mathcal{C} \to \{0, 1\}$, such that each variable $c \in \mathcal{C}$ is assigned the value $o(c)$. Considering these variables, the model likelihood is expressed in the form of a conditional distribution, tailored from the generic form of a CRF [Elk08]:

$$Pr(C = o \mid D, S; W) = \frac{1}{Z} \prod_{\pi=\{c,d,s\}\in\Pi} \phi(c = o(c), d, s; W_\pi) \tag{4.1}$$

where Π is the set of all cliques in the CRF; c, d, s are the claim, document, and source of a clique π, respectively; $Z = \sum_{c\in\mathcal{C}} \prod_{\pi\in\Pi} \phi(c = o(c), d, s; W_\pi)$ is a normalisation

constant to ensure that the probabilities over all configurations of C sum up to one; and $W = \bigcup_{\pi \in \Pi} W_\pi$ is the set of model parameters controlling the effects of individual features. Using this model, we shall compute the conditional distribution of C, given the source and document features. This is realised by the log-linear model (aka logistic regression) that expresses the log of a potential function as a linear combination of features, instantiated from its generic form [Elk08]:

$$\log\ \phi(c = o(c), d, s; W_\pi) = w_{\pi,o(c)} + \sum_{t=1}^{m_\mathcal{D}} w^\mathcal{D}_{\pi,t} \times f^\mathcal{D}_t(d) + \sum_{t=1}^{m_\mathcal{S}} w^\mathcal{S}_{\pi,t} \times f^\mathcal{S}_t(s). \quad (4.2)$$

Hence, we have different weights for each configuration of C and $W_\pi = \{w_{\pi,0}, w_{\pi,1}, w^\mathcal{D}_{\pi,t}, w^\mathcal{S}_{\pi,t}\}$ is the set of all weights.

The above formulation is motivated by the CRF being a special case of log-linear models, which, extending logistic regression, are suitable for structured learning tasks [KBB80, Elk08]. In our setting, the data has an internal structure via the relations between sources, documents, and claims. Exploiting these relations, however, means that the inference of model parameters becomes complex. Hence, the potential function needs to be computationally efficient to enable user interactions in the validation process. A log-linear model enables efficient computation, while, at the same time, provides a comprehensive model, in which the features of sources and documents are discriminative indicators for the credibility of the related claims. The weights enable tuning of feature importance, as features vary between applications and shall be learned from labelled data.

Handling opposing stances. Documents may link the same claim with opposite stances—support or refute it [HN14]—and a source is considered trustworthy, if it refutes an incorrect claim. A model that only captures that a claim is part of a document would neglect this aspect. Yet, incorporating such information via a new type of random variable would mean that the number of variables is larger than or equal to the number of documents, which is much larger than the number of claims (see Section 4.8). We therefore introduce an opposing variable $\neg c$ for each claim c. Then, model complexity increases only slightly: Configurations of C include opposing claims, W contains a doubled number of parameters, and any document connects only to the positive or negative variable of a claim. As c and $\neg c$ cannot have the same credibility value, we enforce a non-equality constraint:

$$Pr(c, \neg c') = \begin{cases} 0 & \text{if } c = c' \\ Pr(c, \neg c' | D, S; W) & \text{otherwise.} \end{cases} \quad (4.3)$$

4.3.2 Incremental Inference with User Input

Using the above formalisation, we further distinguish the set $\mathcal{C}^L \subseteq \mathcal{C}$ of validated, or labelled, claims. It contains all claims c for which, based on user input, we set $P(c) = 1$ in the probabilistic fact database. In the same vein, $\mathcal{C}^U = \mathcal{C} \setminus \mathcal{C}^L$ is the set of unlabelled claims. Based thereon, we define restricted variants of the categorical random variable C that represents credibility configurations of claims: C^U and C^L are variables for configurations involving solely the unlabelled claims of $\mathcal{C}^U$ or the labelled claims of $\mathcal{C}^L$, respectively. Then, we need to solve the following optimisation problem to infer model parameters (as usual, $Pr(X)$ is the probability of one value of a categorical random

variable X), derived from the principle of maximum likelihood [Elk08]:

$$W^* = \arg\max_W \log Pr(C^L \mid D, S; W) \tag{4.4}$$

$$= \arg\max_W \log \sum_{C^U} Pr(C^L, C^U \mid D, S; W). \tag{4.5}$$

The log-likelihood optimisation is convex, since the logarithm is monotonically increasing and the probability distribution is in exponential form. However, the problem becomes intractable due to the exponential number of configurations to consider for the random variable C^U. Moreover, upon receiving new user input, $\mathcal{C}^L$ and $\mathcal{C}^U$, and hence C^L and C^U change, so that re-computation is needed.

Requirements for model inference. To be useful in our setting, an inference algorithm must meet two requirements. First, user input on correspondences should be a first class citizen. By propagating which claims have been validated, credibility probabilities can be computed for claims for no input has been sought so far. Second, each iteration of the validation process changes the credibility of claims only marginally. Hence, inference should proceed incrementally and avoid expensive re-computation of the credibility probabilities and model parameters in each iteration.

Existing inference algorithms. Various inference algorithms have been proposed in the literature. Yet, none of them meets the aforementioned requirements. Traditional CRF models, such as [PMSW17], operate in a static manner, in which model parameters are inferred from a fixed set of labelled data by methods that incur high computational effort (e.g., gradient descent or trusted region methods). Hence, credibility probabilities and model parameters in our model would be computed from scratch every time new user input arrives. Moreover, the instantiation of a grounding based on this model requires another pass over the whole data. This makes it not suitable for interactive validation process considered in our work.

***iCRF* algorithm.** In the light of the above, we propose a novel incremental inference algorithm, *iCRF*, which adopts the view maintenance principle by maintaining a set of Gibbs samples over time. Estimation of credibility and model parameters exploits the results of the previous iteration of the validation process, thereby avoiding re-computation. As we will show experimentally, this does not only increase inference efficiency, but also yields a better approximation compared to random estimation.

Our *iCRF* algorithm implements the third step of the validation process introduced in Section 4.2.3, i.e., the inference of the implications of user input on the probabilistic credibility model. In the z-th iteration of the validation process, reasoning is based on the probabilistic fact database of the previous iteration and the user input that has been received in the z-th iteration. That is, if c is the claim validated in the z-th iteration, we rely on the probabilistic fact database $Q_{z-1} = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P_{z-1} \rangle$, with $\mathcal{C}^U_{z-1}$ and $\mathcal{C}^L_{z-1}$ being the sets of unlabelled and labelled claims, respectively, as indicated by P_{z-1}. Then, these sets are updated, $\mathcal{C}^U_z = \mathcal{C}^U_{z-1} \setminus \{c\}$ and $\mathcal{C}^L_z = \mathcal{C}^L_{z-1} \cup \{c\}$, and inference returns a new probabilistic fact database $Q_z = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P_z \rangle$.

In each iteration of the validation process, our *iCRF* algorithm adopts the Expectation-Maximization (EM) principle for inference. This choice is motivated by EM's fast convergence, computationally efficiency, and particular usefulness when the likelihood is an exponential function (i.e., maximising log-likelihood becomes maximising a linear function). Specifically, we infer the values of the variables for unlabelled claims $\mathcal{C}^U$ through a configuration of C^U and learn the weight parameters W. By relying on an EM-based approach, we can further naturally integrate user input on the credibility of specific claims.

This is a major advantage compared to approaches based on gradient-descent [MW15] that optimise model parameters, but do not enable the integration of user input and constraints (e.g., on opposing claims).

Inference alternates between an *Expectation* (E-step) and a *Maximization* (M-step), until convergence. EM-based inference is conducted in each iteration of the validation process, while each EM iteration updates the model parameters W. Hence, in the z-th iteration of validation, we obtain sequences $W_z^0, W_z^1, \ldots, W_z^l$ and $P_z^0, P_z^1, \ldots, P_z^l$ of model parameters and credibility probabilities.

E-step: We estimate the credibility probabilities from the current parameter values. The first E-step of the z-th iteration of the validation process is based on parameters W_z^0, given as input from the previous iteration of the validation process, i.e., $W_z^0 = W_{z-1}^{l_{z-1}}$, with l_{z-1} as the number of EM iterations in the $z-1$-th iteration of the validation process. In the l-th E-step of the z-th step of the validation process, credibility probabilities are computed as follows:

(1) A sequence of samples Ω_z^l is obtained by Gibbs sampling according to the conditional probability distribution:

$$q_z^l(C_z^U) = Pr(C_z^U \mid C_z^L, D, S; W_z^l) \propto \prod_{\pi=\{c,d,s\}\in\Pi} Pr_z^{l-1}(c) \times \phi(o(c), d, s; W_z^l). \quad (4.6)$$

We incorporate non-equality constraints (Equation 4.3) into Gibbs sampling using an idea similar to [Sch09], which, based on matrix factorisation, embeds constraints as factorised functions into the Markov chain Monte Carlo process. Note that Ω_z^l is a sequence, as any configuration of C^U can appear multiple times. We weight the influence of causal interactions (i.e., cliques) by the credibility of their contained claims, so that user input is propagated via mutual interactions between the cliques.

(2) The probability for each claim $c \in \mathcal{C}^U$ without user input is determined by the ratio of Gibbs samples in which c is credible:

$$Pr_z^l(c) = \frac{\sum_{\omega\in\Omega_z^t} \omega(c)}{|\Omega_z^t|}. \quad (4.7)$$

For all other claims $c \in \mathcal{C}^L$, the probability is fixed by the user input: We set $Pr_z^l(c) = 1$, if the user confirms a claim, and $Pr_z^l(c) = 0$ otherwise.

M-step: We compute the new parameter values by maximising the expectation of log-likelihoods as a weighted average of the probability distribution of current label estimates. That is, in the l-th M-step of the z-th step of the validation process, we have:

$$W_z^{l+1} = \arg\max_{W'} \sum_{C^U} q_z^l(C_z^U) \log Pr(C_z^L, C_z^U | D, S; W') \quad (4.8)$$

This step is realised by a L2-regularized Trust Region Newton Method [LWK08], suited for large-scale data, where critical information is often sparse (many zero-valued features).

Proposition 3. iCRF *runs in linear time in the size of the dataset.*

Proof. The E-step is implemented by Gibbs sampling, which takes linear time [CG92, KN+99] in the number of claims. The M-step is implemented by the Trust Region Newton Method, which also takes linear time in the dataset size [LWK08] . □

4.3.3 Instantiation of a Grounding

Once the user input of the z-th iteration of the validation process has been incorporated, a grounding is instantiated. This corresponds to the fourth step of the validation process in Section 4.2.3, i.e., deciding which claims are deemed credible. Since claims are not independent, we take the configuration with maximal joint probability:

$$g_z(c) = \begin{cases} 1 & \text{if } (c \in \mathcal{C}_z^L) \vee \\ & (o(c) = 1 \wedge o = \arg\max_{C_z^U} Pr(C_z^U \mid C_z^L, D, S; W_z)) \\ 0 & \text{otherwise.} \end{cases} \tag{4.9}$$

However, solving this equation is similar to solving a Boolean satisfiability problem. Thus, we simply leverage the most recent Gibbs sampling result Ω_z^*, obtained during EM, for instantiation. This is defined by a function *decide* as follows:

$$\begin{aligned} g_z(c) &= decide(c, \Omega_z^*) \\ &= \begin{cases} 1 & \text{if } (c \in \mathcal{C}_z^L) \vee \\ & (o(c) = 1 \ \wedge o = \arg\max_{C_z^U} |\{\omega \in \Omega_z^* \mid C_z^U = \omega\}|) \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{4.10}$$

Consider a set of claims $\mathcal{C} = \{c_1, c_2, c_3\}$ and assume that the last Gibbs sampling comprised three configurations, $\omega_1 = [1, 1, 0]$, $\omega_2 = [1, 0, 0]$, $\omega_3 = [1, 1, 0]$, where the i-th vector element denotes the credibility of claim c_i. Instantiation will return $[1, 1, 0]$ as this configuration appears most often, so that its probability is maximal.

4.4 User Guidance

Having discussed (i) inference based on user input and (ii) instantiation of a grounding, we turn to strategies to guide a user in the validation. This corresponds to the first step of the validation process presented in Section 4.2.3, i.e., the selection of a claim for validation. We first define a measure of uncertainty for a probabilistic fact database (Section 4.4.1). Then, two selection strategies are introduced (Section 4.4.2 and Section 4.4.3), before they are combined in a hybrid approach (Section 4.4.4).

4.4.1 Uncertainty Measurement

The model of a probabilistic fact database, as constructed above, enables us to quantify the uncertainty related to credibility inference in order to guide a user in the validation process. Let $Q = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P \rangle$ be a probabilistic fact database. Recall that P assigns to each claim $c \in \mathcal{C}$ the probability $P(c)$ of it being credible, while C is the categorical random variable that captures credibility configurations over all claims. We quantify the overall uncertainty of the database by the Shannon entropy over a set of claims:

$$H_C(Q) = -\sum_C Pr(C; W) \log Pr(C; W) \tag{4.11}$$

In our iCRF model, it can be computed exactly by [RN09, Rey13]:

$$H_C(Q) = \Phi(W) - \mathbb{E}_W[t(C)]^T W \tag{4.12}$$

where $\Phi(W) = \sum_C \prod_\pi \phi(o, d, s; W)$ is called the partition function and $\mathbb{E}_W[t(C)] = \nabla\Phi(W)$. Since our model is an acyclic graph with no self statistics, the partition function is computed exactly using Ising methods [Rey13], which run in polynomial time.

We can further scale-up uncertainty computation by approximating the entropy in linear time, as follows:

$$H_C(Q) = -\sum_{c \in \mathcal{C}} [Pr(c) \log Pr(c) + (1 - Pr(c)) \log(1 - Pr(c))] \tag{4.13}$$

where the claim probabilities are obtained after each EM iteration (i.e., Equation 4.7 for unlabelled claims, or directly by the user input for labelled claims). However, this approximation neglects the mutual dependencies between claims.

4.4.2 Information-driven User Guidance

A first heuristic to guide the selection of claims for validation aims at the maximal reduction in uncertainty under the assumption of trustworthy sources. It exploits the benefit of validating a claim using the notion of information gain from information theory [RN03].

To capture the impact of user input on a claim c, we define a conditional variant of the entropy measure introduced earlier. It measures the expected entropy of the database under specific validation input:

$$H_C(Q \mid c) = Pr(c) \times H_C(Q^+) + (1 - Pr(c)) \times H_C(Q^-) \tag{4.14}$$

where $Q^+ = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P^+ \rangle$ and $Q^- = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P^- \rangle$ are inferred from $Q = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P \rangle$ by iCRF (Section 4.3.2), under input that confirms the claim, $P^+(c) = 1$, or labels it as non-credible, $P^-(c) = 0$.

To take a decision on which claim to select, we assess the expected difference in uncertainty before and after incorporating input for a claim. The respective change in entropy is the information gain that quantifies the potential benefit of knowing the true value of an unknown variable [RN03], i.e., the credibility value in our case:

$$IG_C(c) = H_C(Q) - H_C(Q \mid c). \tag{4.15}$$

Using this notion, we chose the claim that is expected to maximally reduce the uncertainty of the probabilistic fact database. This yields a selection function for information-driven user guidance:

$$select_C(\mathcal{C}) = \arg\max_{c \in \mathcal{C}} \ IG_C(c) \tag{4.16}$$

Note that we do not need to rank the opposing claim $\neg c$ of a claim c, as their conditional entropies in Equation 4.14 will be equivalent.

4.4.3 Source-driven User Guidance

User guidance as introduced above assumes that sources are trustworthy—an assumption that is often violated in practice. To tackle this issue, we model source trustworthiness by explicitly aggregating over all claims made by a source. More precisely, the likelihood that a source is trustworthy is measured as the fraction of its claims that are considered credible. The latter is derived from the grounding g_z instantiated in the last, the z-th, EM iteration:

$$Pr(s) = \frac{\sum_{c \in \mathcal{C}_s} g_z(c)}{|\mathcal{C}_s|} \tag{4.17}$$

where $\mathcal{C}_s = \{c \in \mathcal{C} \mid (c, s) \in \Pi\}$ is the set of claims connected to s in the CRF model. Then, the uncertainty of source trustworthiness values is defined as:

$$H_S(Q) = -\sum_{s \in \mathcal{S}} [Pr(s) \log Pr(s) + (1 - Pr(s)) \log(1 - Pr(s))] \tag{4.18}$$

The conditional entropy when a claim c is validated is:

$$H_S(Q|c) = Pr(c) \times H_S(Q^+) + (1 - Pr(c)) \times H_S(Q^-) \tag{4.19}$$

where, as detailed above, Q^+ and Q^- are inferred from Q by iCRF under user input that confirms or disproves the claim, i.e., setting $P^+(c) = 1$ for Q^+, or $P^-(c) = 0$ for Q^-, respectively.

As for the first heuristic, we further capture the information gain as the difference in entropy and, based thereon, define the selection function for source-driven user guidance:

$$IG_S(c) = H_S(Q) - H_S(Q|c) \tag{4.20}$$

$$select_S(\mathcal{C}) = \arg\max_{c \in \mathcal{C}} IG_S(c) \tag{4.21}$$

Again, we do not need to rank opposing claims.

4.4.4 Hybrid User Guidance

There is a trade-off between the information-driven and the source-driven strategy for user guidance. Focusing solely on the former may lead to contamination of the claims from trustworthy sources by unreliable sources. An excessively source-driven approach, in turn, may increase the overall user efforts significantly. Thus, we propose a dynamic weighting procedure that to choose among the two strategies. This choice is influenced by two aspects:

Ratio of untrustworthy sources. If there is a high number of unreliable sources, the source-driven strategy is preferred. With little user input, detection of unreliable sources is difficult, though, so that the information-driven strategy is favoured in the beginning.

Error rate. The grounding g_i captures which claims are deemed credible in the i-th iteration of the validation process. If g_i turns out to be mostly incorrect, we have evidence of unreliable sources and favour the source-driven strategy.

Initially, with little user input, we choose the strategy mainly based on the error rate of the grounding. At later stages of the validation process, the number of inferred unreliable sources becomes the dominant factor. The above idea is formalised based on the ratio of unreliable sources in the i-th iteration of the validation process, which is $r_i = (|\{s \in \mathcal{S} \mid Pr(s) < 0.5\}|)/(|\mathcal{S}|)$. The error rate of the grounding is computed by comparing the user input for claim c in the i-th iteration with the credibility value assigned to c in g_{i-1}, i.e., in the previous iteration. Here, we leverage the probability $P_{i-1}(c)$ of the probabilistic fact database $Q_{i-1} = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P_{i-1} \rangle$, of the previous iteration. The error rate is computed as:

$$\epsilon_i = \begin{cases} 1 - Pr_{i-1}(c) & g_{i-1}(c) = 1 \\ Pr_{i-1}(c) & \text{otherwise} \end{cases} \tag{4.22}$$

Using the ratio of unreliable sources r_i and the error rate ϵ_i, a we define a score for choosing the source-driven strategy:

$$z_i = 1 - e^{-(\epsilon_i(1-h_i)+r_i h_i)} \tag{4.23}$$

where $h_i = i/|\mathcal{C}|$ is the ratio of user input. This score mediates the trade-off between the error rate ϵ_i and the ratio of untrustworthy sources r_i by the ratio of user input h_i. When the ratio h_i is small, the ratio of untrustworthy sources has less influence and the error rate is the dominant factor. When the ratio h_i is large, the ratio of unreliable sources becomes a more dominant factor.

4.5 Complete Validation Process

Combining the techniques for credibility inference and instantiation of a grounding (Section 4.3) with those for user guidance (Section 4.4), we define a comprehensive validation process (Section 5.4). We further outline how robustness against erroneous user input is achieved (Section 4.5.2).

4.5.1 The Algorithm

Our complete validation process for fact checking is defined in Algorithm 4.1. It instantiates the general validation process outlined in Section 4.2.3 to address the problem of effort minimisation (Problem 2). As long as the validation goal is not reached and the user effort budget has not been exhausted (line 6), selection of the claim for which user input shall be sought is done either by the source-driven or the information-driven strategy. The choice between strategies is taken by comparing factor z_{i-1} to a random number (line 8), which implements a roulette wheel selection. The second step (lines 10-13) elicits user input for the selected claim and computes the error rate. The third step incorporates the user input in the probabilistic model (line 14) and then conducts credibility inference by means of our iCRF algorithm (line 15). This yields a new probabilistic model P_i, along with the Gibbs sampling result Ω_i^* of the last E-step. Based thereon, in a fourth step, we decide on the new grounding g_i capturing the facts that are considered credible (line 16). The ratio of unreliable sources r_i is calculated to compute score z_i (lines 17-18), used in the next iteration to choose between the selection strategies.

Proposition 4. *An iteration of Algorithm 4.1 (lines 6-19) runs in linear time in the size of the dataset.*

Proof. The time complexity of the iteration of Algorithm 4.1 is dominated by the *iCRF* algorithm, which infers the implications of new user input. Yet, *iCRF* runs in linear time in the dataset size (Proposition 3). □

Applying Algorithm 4.1 in practice, the computation of the information gain for the information-driven or source-driven selection strategy becomes a performance bottleneck. Therefore, we consider two optimisations for this step:

- *Parallelisation:* The computation of information gain for different claims is independent and thus done in parallel.
- *Graph partitioning:* Not all sources share the same claims and not all claims stem from a single source. Hence, as a pre-processing step before seeking user input, the graph representation of the CRF model can be decomposed into its connected components [KFL01]. The resulting smaller CRF models can then be handled more efficiently.

4.5.2 Robustness Against User Errors

When validating claims, a user may make mistakes, not because of a lack of knowledge, but as a result of the interactions with a validation system [Rea90]. Assuming that a user is confronted with the current inferred credibility of the claim to validate, along with an assessment of related sources and documents, any decision to deviate from the current most likely credibility assignment is typically taken well-motivated. Common

Algorithm 4.1: Validation process for fact checking

input : sets of sources $\mathcal{S}$, documents $\mathcal{D}$, and claims $\mathcal{C}$,
$\mathcal{C}_s \subseteq \mathcal{C}$ being claims originating from a source $s \in \mathcal{S}$,
a validation goal Δ, and a user effort budget b.
output: the grounding g.

$\mathcal{C}^U \leftarrow \mathcal{C}$; $\mathcal{C}^L \leftarrow \emptyset$;
$(P_0, \Omega_0^*) \leftarrow iCRF(\mathcal{S}, \mathcal{D}, \mathcal{C}, (c \mapsto 0.5, c \in \mathcal{C}))$;
$g_0 \leftarrow (c \mapsto decide(c, \Omega_0^*), c \in \mathcal{C})$;
$z_0 \leftarrow 0$;
$i \leftarrow 1$;
while *not* $\Delta \ \wedge \ i < b$ **do**
  `// (1) Select a claim to validate`
  $x \leftarrow random(0, 1)$;
  `// Source-driven or information-driven strategy?`
  if $x < z_{i-1}$ **then** $c \leftarrow select_S(\mathcal{C}^U)$;
  else $c \leftarrow select_C(\mathcal{C}^U)$;
  `// (2) Elicit user input`
  Elicit user input $v \in \{0, 1\}$ on c;
  $\mathcal{C}^U \leftarrow \mathcal{C}^U \setminus \{c\}$; $\mathcal{C}^L \leftarrow \mathcal{C}^L \cup \{c\}$;
  `// Calculate error rate` ϵ_i
  if $g_{i-1}(c) = 1$ **then** $\epsilon_i = 1 - P_{i-1}(c)$;
  else $\epsilon_i = P_{i-1}(c)$;
  `// (3) Infer implications of user input`
  `// Update credibility of validated claim`
  $P \leftarrow (c \mapsto v \ \wedge \ c' \mapsto P_{i-1}(c'), c' \in \mathcal{C}, c' \neq c)$;
  `// Conduct inference`
  $(P_i, \Omega_i^*) \leftarrow iCRF(\mathcal{S}, \mathcal{D}, \mathcal{C}, P)$;
  `// (4) Decide on grounding`
  `// Instantiate grounding based on samples of last iCRF`
  $g_i \leftarrow (c \mapsto decide(c, \Omega_i^*), c \in \mathcal{C})$;
  `// Calculate ratio of unreliable sources`
  $r_i = \frac{1}{|\mathcal{S}|} \left| \left\{ s \in \mathcal{S} \mid \frac{\sum_{c \in \mathcal{C}_s} g_i(c)}{|\mathcal{C}_s|} < 0.5 \right\} \right|$;
  `// Calculate score to choose selection strategy`
  $z_i = 1 - e^{-\left(\epsilon_i \left(1 - \frac{i}{|\mathcal{C}|}\right) + r_i \frac{i}{|\mathcal{C}|}\right)}$;
  $i \leftarrow i + 1$;
return g_{i-1}

mistakes, thus, are accidental confirmations of a (wrong) inferred credibility value of a claim.

Against this background, we incorporate a lightweight confirmation check, triggered after a fixed number of iterations of the validation process. At some step i, for every claim c that has been validated, a grounding $g^i_{\sim c}$ is constructed, using all information of the probabilistic fact database except the validation of c. Then, the label for claim c in $g^i_{\sim c}$ is compared with the respective user input v. If $g^i_{\sim c}(c) \neq v$, then v is identified as a potential mistake and updated accordingly. Intuitively, this check exploits that additional user input may lead to a different inferred credibility value than the one given earlier directly by the user. As inference is based on a large number of validated claims, instead of a single one, it is considered more trustworthy. We will demonstrate experimentally that this check is highly effective when trying to detect user mistakes.

4.6 Methods for Effort Reduction

Based on the validation process introduced so far, this section presents methods to further reduce the required user effort. Detecting convergence of our probabilistic model, we discuss when to terminate validation (Section 4.6.1). Reducing set-up costs of a user,

we then target batching of claims (Section 4.6.2).

4.6.1 Early Termination

In practice, we can improve efficiency by terminating the validation process upon convergence of the results. Below, we define several criteria that indicate such convergence and, therefore, may be employed as additional termination criteria.

Uncertainty reduction rate. A first indicator is the effect of user input in terms of uncertainty reduction. After each iteration in Algorithm 4.1, the probabilistic fact database Q_i becomes Q_{i+1}. The rate of uncertainty reduction is measured as $\frac{(H_C(Q_i)-H_C(Q_{i+1}))}{H_C(Q_i)}$. The rate approaches zero upon convergence, so that validation is stopped once the rate falls below a threshold.

The amount of changes. Instead of considering the probability values of all claims, this indicator incorporates solely the configuration with the highest likelihood. With g_i and g_{i+1} as the groundings of two iterations of Algorithm 4.1, the amount of change is quantified as $|\{c \in \mathcal{C} \mid g_i(c) \neq g_{i+1}(c)\}|$. If this value becomes negligible, i.e., falls below a threshold over several consecutive iterations, we conclude that the credibility of claims has been determined.

Amount of validated predictions. Another indicator for a high quality model is the ability to instantiate credibility assignments that are matched with user input. Exploiting this idea, in each iteration of Algorithm 4.1, we assess whether the result of inference and the user input are consistent. If this is the case for several consecutive iterations, we conclude that the validation process may be stopped.

Precision improvement rate. A more direct way to assess convergence is to estimate the precision based on k-fold cross validation. Formally, in the i-th iteration of Algorithm 4.1, the set of labelled claims $\mathcal{C}^L$ is divided into k equal-size partitions, $E = E_1 \cup \ldots \cup E_k$. Then, we repeat the following procedure k-times: (1) consider the claims of the j-th partition E_j as non-validated; (2) conduct credibility inference ignoring the user input for claims in E_j and instantiate a grounding g'_j; (3) compare the credibility values for claims in E_j based on g'_j with those given directly by the user: $A_{E_j} = (|\{c \in E_j \mid g'_j(c) = P_i(c)\}|)/|E_j|$. We then take the average of k runs as an overall estimation of the model precision at step i, i.e., $A_i = (\sum_{j=1}^{k} A_{E_j})/k$. This yields a rate $(A_i - A_{i-1})/A_{i-1}$ of precision improvement at step i. This rate shall converge to zero, thereby indicating when to terminate the validation process.

4.6.2 Batching

Batching of claims reduces the set-up costs of users, i.e., the time needed to familiarise with a particular domain. Moreover, batching enables the definition of large validation tasks, which is beneficial when involving multiple users working in parallel. We thus adapt the approach defined in Algorithm 4.1, so that a set of claims, instead of a single one, is checked per iteration. Below, we show how to lift claim selection to sets, assessing the benefit of their joint validation.

Expected benefit. We measure the information gain of validating claims $\mathcal{B} \subseteq \mathcal{C}$ by the expected uncertainty reduction. With B as the categorical random variable that represents credibility configurations of claims $\mathcal{B}$, the uncertainty conditioned by user input on $\mathcal{B}$ is:

$$H_C(Q \mid B) = \sum_{B} Pr(B) H(Q^B) \tag{4.24}$$

Here, Q^B denotes the probabilistic fact database constructed after incorporating the given configuration of B. Note that a more complex cost model could be constructed based on validation difficulty (e.g., implied by logical relations between claims) [BRLT+15]. Yet, this is orthogonal to our work. Using this measure, our validation process incorporates batching of claims by choose the top-k claims with maximal information gain (breaking ties randomly):

$$select_B(\mathcal{C}) = \underset{\mathcal{B}\subseteq\mathcal{C},|\mathcal{B}|=k}{\arg\max}\ H_C(Q) - H_C(Q \mid B) \tag{4.25}$$

However, the above optimisation problem is computationally hard, as, in practice, both $|\mathcal{C}|$ and k are large. We therefore resort to an *approximate computation* of the benefit and a *greedy algorithm* for the actual selection.

Approximating the expected benefit. We employ an alternative utility function that combines the individual benefit of each claim with a redundancy penalty that incorporates claim dependencies.

Individual benefit: The expected benefit of a claim c is computed as its information gain $IG_C(c)$ as defined in Equation 4.15, which is tractable. Selecting claims one-by-one based solely on their individual benefit, however, may be non-optimal, due to the complex joint distribution of random variables for claims, documents, and sources.

Redundancy penalty: Neglecting the dependencies between the variables in the CRF model may yield redundant validation effort. Therefore, when selecting claims, we aim at low information overlap, which is quantified as the redundancy of a set of claims $\mathcal{B} \subseteq \mathcal{C}$ as:

$$R(\mathcal{B}) = \sum_{c,c'\in\mathcal{B}} IG_C(c)M(c,c')IG_C(c') \tag{4.26}$$

where $M(c,c') = \frac{1}{Z}|\{s \in S | c \in \mathcal{C}_s \wedge c' \in \mathcal{C}_s\}|$ is a correlation matrix that is based on the number of sources that serve as the origin of both claims c and c' and normalised to the unit interval by $Z = \max_{c,c'\in C} M(c,c')$.

Approximated benefit: The two aforementioned measures are combined to approximate the benefit of validating a set of claims $\mathcal{B} \subseteq \mathcal{C}$. The individual benefit, however, is weighted by the importance of a claim. The idea is that claims stemming from a large group of dependent claims have a high chance to propagate information. To exploit this effect, we define $q(c) = \sum_{c'\in\mathcal{C}} M(c,c')IG_C(c')$ as the importance of claim c. Putting it all together, we employ the following utility function to approximate the benefit of validating $\mathcal{B}$:

$$F(\mathcal{B}) = w\sum_{c\in\mathcal{B}} q(c)IG_C(c) - \sum_{c,c'\in\mathcal{B}} IG_C(c)M(c,c')IG_C(c') \tag{4.27}$$

where $w \in \mathbb{R}^+$ is a positive weight parameter to balance the terms related to individual benefit and redundancy. Then, our utility function is used to guide the selection of the top-k claims:

$$select_{AB}(\mathcal{C}) = \underset{\mathcal{B}\subseteq\mathcal{C},|\mathcal{B}|=k}{\arg\max}\ F(\mathcal{B}) \tag{4.28}$$

As discussed, computation of the utility function F is tractable. However, the above optimisation problem (Equation 4.28) is not.

Theorem 4.6.1. *Computing the result of $select_{AB}$ is NP-complete.*

Proof. F is a submodular set function. Maximization of such functions is known to be NP-complete [NW81]. □

Greedy selection. Exploiting the monotonicity and submodularity of the utility function F, we define a greedy algorithm with a performance guarantee of $(1 - 1/e) \approx 0.63$ [NW81]. We iteratively expand the set of claims in k iterations. In each iteration, we traverse all unlabelled claims to identify the claim c^* to maximise the gain $\Delta(c^*) = F(\mathcal{B}' \cup \{c^*\}) - F(\mathcal{B}')$, where $\mathcal{B}'$ is the set of claims selected in the previous iteration. Note that the gain can be updated incrementally. That is, $\Delta_{i+1}(c) = \Delta_i(c) - 2IG_C(c_i^*)M(c, c_i^*)IG_C(c)$, where c_i^* is the claim chosen in iteration i.

The time and space complexity of this heuristic strategy are $\mathcal{O}(|\mathcal{C}|^2+k|\mathcal{C}|)$ and $\mathcal{O}(|\mathcal{C}|^2)$, respectively. The quadratic term $|C|^2$ in either complexity stems from the calculation of the correlation matrix $M(.,.)$. The linear term $k|C|$ is explained by k iterations, each requiring consideration of the whole set of claims to select c^*.

4.7 Streaming Fact Checking

We now lift our approach to a streaming setting. Instead of checking a large set of claims from scratch, we consider a potentially infinite stream of claims to validate.

Upon the arrival of new documents, sources, and claims, the model structure and its parameters need to be updated. However, evaluating the parameters periodically based on the complete database is not a viable option, as the database grows continuously. Limiting the number of considered claims, in turn, may induce a loss of all claims provided by a source. Since only a (small) subset of documents is observed per source, operating on a subset of claims increases the risk of discarding trustworthy sources and documents.

We therefore propose an online expectation-maximization algorithm that reuses and updates the previous trained parameters, which accelerates convergence in the presence of new data. We operate on one claim at a time, and both the claim and the associated user input are discarded after validation. As such, we can only provide an educated guess on the credibility of the claim at a later stage. However, this is a minor drawback, since, in an online setting, claims are relevant only for a comparatively short interval. How to decide on which claims to discard in a more elaborated manner, is an interesting problem, see [NDW+17], yet orthogonal to our work.

Algorithm 4.2: Streaming fact checking

input : Probabilistic fact database $Q = \langle \mathcal{S}, \mathcal{D}, \mathcal{C}, P \rangle$ and its CRF representation $Pr(C|D, S; W)$,
A potentially infinite stream of claims $c_1, c_2, \ldots$.

while *a new non-validated claim* c_t *arrives* **do**
  $C_t^U \leftarrow C_{t-1}^U \cup \{c_t\}$;
  if c_t *comes with a new document* d_t **then** $D_t \leftarrow D_{t-1} \cup \{d_t\}$;
  else $D_t \leftarrow D_{t-1}$;
  if c_t *comes with a new source* s_t **then** $S_t \leftarrow S_{t-1} \cup \{s_t\}$;
  else $S_t \leftarrow S_{t-1}$;
  Receive current model parameters W from Algorithm 4.1 ;
  Compute $Q_t(W)$ using Equation 4.29 ;
  Compute W_t using Equation 4.30 ;
  Feed new model parameters W_t to Algorithm 4.1 ;

In the online setting, we consider an EM algorithm with stochastic approximation to update the likelihood with a new claim c_t, a new source s_t, or a new document d_t,

rather than conducting re-computation. Specifically, the update rule is defined as:

$$Q_t(W) = Q_{t-1}(W) + \gamma_t \times \left(\mathop{\mathbb{E}}_{C_t^U | C_t^L, D_t, S_t, W_{t-1}} [\log Pr(C_t^U, C_t^L, D_t, S_t; W)] - Q_{t-1}(W) \right) \tag{4.29}$$

where $Q_0(W) = 0$ and the sequence $\gamma_1, \gamma_2, \ldots$ is a decreasing sequence of positive step sizes, i.e. $\lim_{T\to\infty} \sum_{t=1}^{T} \gamma_t = \infty$ and $\lim_{T\to\infty} \sum_{t=1}^{T} \gamma_t^2 < \infty$. In practice, the step-size y_t may be adjusted using line searches to ensure that the likelihood is indeed increased in each iteration [CM09]. As above, the model parameters W are estimated by maximizing the expectation of the likelihood via the L2-regularized Trust Region Newton Method [LWK08]:

$$W_t = \arg\max_W Q_t(W) \tag{4.30}$$

We realise this idea in Algorithm 5.1. Given a stream of claims $c_1, c_2, \ldots$, the algorithm updates the model variables C_t^U, D_t, S_t (lines 2 to 6). It then performs the stochastic approximation of the parameter estimates (lines 8 to 9). The returned parameters are then fed to Algorithm 4.1 (line 10). Algorithm 5.1 can receive the current model parameters from Algorithm 4.1 (line 7), since both algorithms may run in parallel and influence the parameters of one another. The respective parts of either algorithm are highlighted. That is, user input in Algorithm 4.1 or the arrival of a new claim in Algorithm 5.1 may change the model.

Proposition 5. *Algorithm 5.1 runs in linear time.*

Proof. The update of a new claim is implemented by Trust Region Newton Method, which takes linear time [LWK08] in the dataset size. □

4.8 Evaluation

We evaluate our approach experimentally, using real-world datasets. We first discuss the experimental setup (Section 4.8.1), before turning to an evaluation of the following aspects of our approach:

- The runtime performance of the presented approach (Section 4.8.2).
- The efficacy of the CRF model (Section 4.8.3).
- The effectiveness of user guidance (Section 4.8.4).
- The robustness against erroneous user input (Section 4.8.5).
- The effectiveness of early termination (Section 4.8.6).
- The benefits and trade-offs of batching (Section 4.8.7).
- The streaming setting of fact checking (Section 4.8.8).
- The real-world deployment for human validators (Section 4.8.9).

4.8.1 Experimental Setup

Datasets. We utilise state-of-the-art datasets in fact checking [ZL18]:

- *Wikipedia:* The dataset contains proven hoaxes and fictitious people from Wikipedia [Wik17] with 1955 sources, 3228 documents, and 157 labelled claims. The model has been constructed by taking unique, curated claims from Wikipedia and using them as a query for a search engine to collect Web pages as documents, while the originating domain names indicate the sources. The top-30 retrieved documents are linked to a given claim, except those that originate from wikipedia.org in order to avoid a bias, as described in [PMSW17].
- *Healthcare forum:* The dataset contains 291276 claims about side-effects of drugs extracted from 2.8M documents of 15K users on healthboards.com [Dat17a]. We consider 529 claims of 48083 documents from 11206 users, which have been labelled by health experts. The model has been constructed using domain-specific rules to extract RDF triples from forum texts, i.e. documents. Each user of the forum is considered as a source. Various pattern mining and data cleaning routines are used to ensure that the resulting set of claims does not contain duplicates, see [MWDNM14b].
- *Snopes:* This dataset [dat17b] originates from the by far most reliable and largest platform for fact checking [VRA18], covering different domains such as news websites, social media, and e-mails. The dataset comprises 80421 documents of 23260 sources that contain 4856 labelled claims. The model has been constructed as described above for the *wikipedia* dataset: A duplicate-free set of curated claims of the Snopes' editors was used to collect Web pages that links to these claims, see [PMSW17].

For these datasets, we derive features as follows. If a source is a website, we rely on centrality scores such as PageRank and HITS. If a source is an author, features include personal information (age, gender) and activity logs (number of posts). Language quality of documents is assessed using common linguistic features such as stylistic indicators (e.g., use of modals, inferential conjunction) and affective indicators (e.g., sentiments, thematic words) [OPLA13].

We follow common practice [MSF+14, AGK10, HTWA15c, NNM+14, HTT+17] and use the ground truth of the datasets to simulate user input. Model parameters are initialised with 0.5, following the maximum entropy principle.

Evaluation measures. In addition to the uncertainty of a probabilistic fact database, see Section 4.4, we measure:

User effort (E): the ratio of validated claims $|\mathcal{C}^U|$ and all claims $|\mathcal{C}|$, i.e., $E = |\mathcal{C}^U|/|\mathcal{C}|$.

Precision (P_i): the correctness of the grounding. Let $g^* : \mathcal{C} \to \{0, 1\}$ be the correct assignment of credibility values. Then, we measure precision of grounding g_i in the i-th iteration of the validation process as $P_i = |\{c \in \mathcal{C} \mid g_i(c) = g^*(c)\}|/|\mathcal{C}|$. This definition of precision is different from the one in information retrieval and binary classification [RN03]. As the user interest is a trusted set of facts, the correctness of obtained facts is evaluated.

Precision improvement (R_i): a normalised version of precision, measuring relative improvements to illustrate the effect of user input. With P_0 as the initial precision, the measure is defined at the i-th iteration by $R_i = P_i - P_0/1 - P_0$.

Experimental environment. Our results have been obtained on an Intel Core i7 system (3.4GHz, 12GB RAM). All except the experiments on early termination (Section 4.8.6) ran until the actual termination of the validation process.

4.8.2 Runtime Performance

We first measures the response time, denoted by Δt, of our approach during one iteration of Algorithm 4.1, i.e., the wait time of a user. This includes the time for inference and claim selection.

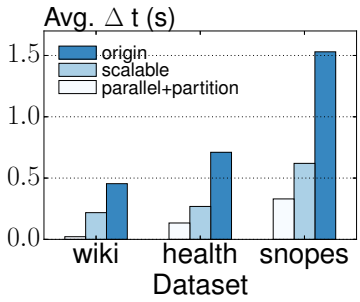


Figure 4.2: Time vs. dataset

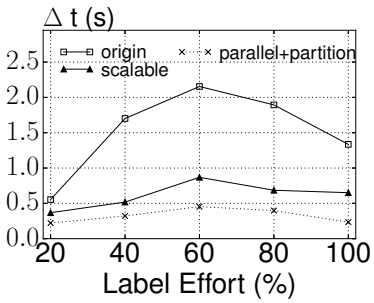


Figure 4.3: Time vs. effort

Figure 4.2 shows the observed response time, averaged over 10 runs, when using the plain algorithm (*origin*), with uncertainty estimation as introduced in Section 4.4.1 (*scalable*), and with the computational optimisations of Section 5.4 (*parallel+partition*). With larger dataset size (*wiki* to *snopes*), the response time increases. However, with computational optimisations, the average response time stays below half a second, which enables immediate user interactions. Figure 4.3 further illustrates for the largest dataset, *snopes*, how the response time evolves during validation when averaging the response time over equal bins of relative user effort. The response time peaks between 40% and 60% of user effort, since at these levels, user input enables the most conclusions on credibility values.

4.8.3 Efficacy of the CRF Model

Next, we assess the estimated probabilities of credibility assignments. Since we use probabilistic information to guide validation, the probabilities should reflect the ground truth, i.e., the true credibility values of claims. For each claim, our model should assign a higher probability to correct credibility values than to incorrect ones. In the experiment, we keep track of the correct assignments (if a claim is correct, we plot $Pr(c = 1)$ and otherwise, we plot $Pr(c = 0)$) and their associated probabilities, while varying the user effort (0%, 20%, 40%).

Figure 4.4 shows a histogram over all datasets, illustrating how often the probability assigned to a claim falls into a specific bin. Increasing the amount of user effort, the range covering most of the correct credibility values shifts from lower probability bins to higher ones. Even with little user effort (20%), the number of correct assignments with a value ≥ 0.5 is high. This highlights that user input indeed enables a better assessment of the credibility of claims.

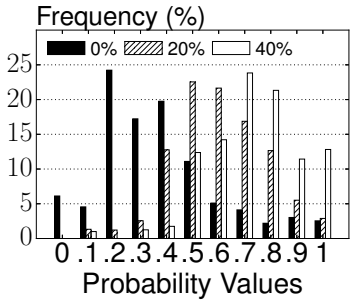


Figure 4.4: Guidance benefits

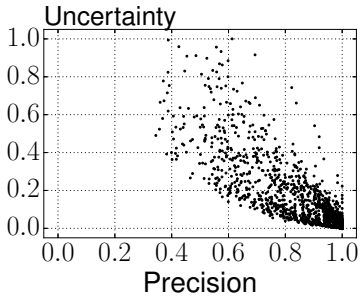


Figure 4.5: Uncert. vs. prec.

4.8.4 Effectiveness of User Guidance

Relation between uncertainty and precision. We verify our assumption that the uncertainty of a fact database, see Section 4.4, is correlated with the precision of the grounding. In this experiment, the information-driven guidance was applied to all datasets (100 runs each), until precision reaches 1.0. Figure 4.5 plots the observed values for precision and normalised uncertainty (i.e., uncertainty divided by the maximum value of the run). There is a strong correlation between both measures (Pearson's coefficient is -0.8523, a highly negative correlation). Hence, uncertainty is indeed a truthful indicator of correctness of the credibility assignments.

Guidance strategies. In this experiment, we mimic the user by the ground-truth, until precision reaches 1.0. We compare our approach (*hybrid*) with four baseline methods: *random*, which selects a claim randomly; *uncertainty*, which selects the most 'problematic' claim, in terms of the entropy of its probability; *info*, which uses the information-driven user guidance only; and *source*, which uses the source-driven user guidance only. Figure 4.6 shows the results for all datasets. Our approach (*hybrid*) clearly outperforms baseline techniques. For example, using the *snopes* dataset, our approach leads to a precision value > 0.9 with input on only 31% of the claims, whereas the other methods require validation of at least 67% to reach the same level of precision.

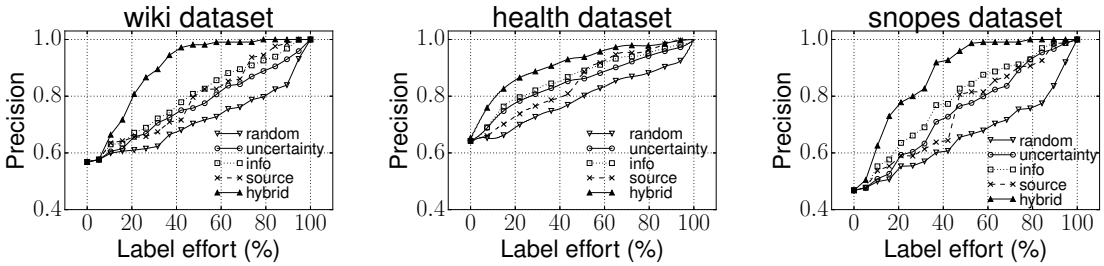


Figure 4.6: Effectiveness of guiding

4.8.5 Robustness Against Erroneous User Input

Detecting erroneous input. We evaluate our approach to detect erroneous input by simulating user mistakes. With a probability p, we transform correct user input into an incorrect assessment.

The confirmation check (Section 4.5.2) is triggered after each 1% of total validations. Table 4.1 shows the detected mistakes (%) when increasing parameter p. Across all

Table 4.1: Detected mistakes

Dataset	p : probability of mistake			
	0.15	0.20	0.25	0.30
wiki	100	100	96	89
health	100	100	94	86
snopes	100	95	87	79

datasets, the majority of inserted mistakes is detected.

User guidance with mistakes. We further study the effect of user mistakes on the relation between user effort and precision. Again, the confirmation check is triggered after each 1% of total validations. Upon a detected mistake, the user reconsiders the input, which adds to the invested effort. Figure 4.7 illustrates that this implies that more user interactions are required to reach perfect precision. However, the precision curves obtained with our approach are still much better than with other baseline methods.

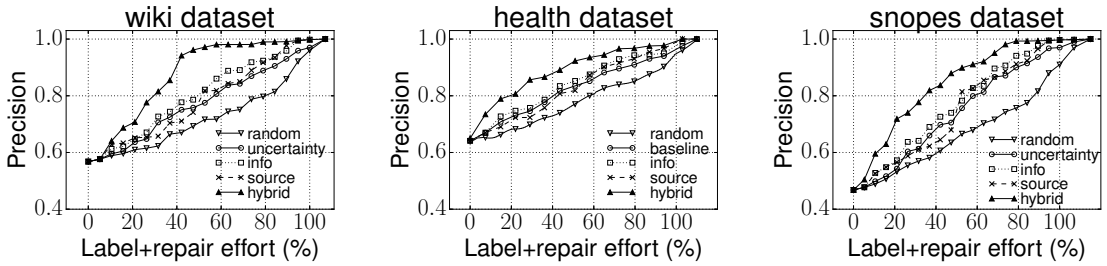


Figure 4.7: Guiding with erroneous user input

Effects of missing user input. A user may skip the validation of a claim due to being unsure or preferring to check another claim first. We consider such scenarios by a probability p_m with which a claim is skipped, meaning that the second-best claim is validated. We test p_m ranging from 0.1 to 0.5, while running the validation process until a precision value of 0.7, 0.8, or 0.9 is reached. Figure 4.8 shows the saved efforts (%), computed as the relative difference in user effort between the normal process and the one with skipping, needed to reach the respective precision. As expected, skipping at the beginning of the validation process (precision level of 0.7) affects the saved effort, as selecting the second-best candidate leads to worse inference results. Later, this effect becomes smaller.

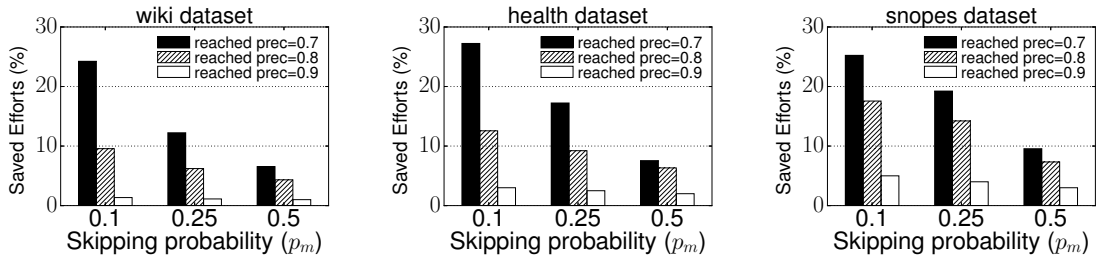


Figure 4.8: Effects of missing user input

4.8.6 Benefits of Early Termination

Using the *snopes* dataset (*wiki* and *health* show similar trends), we evaluate our indicators for early termination of the validation process (see Section 4.6.1): the uncertainty reduction rate (*URR*); the amount of changes (*CNG*); the amount of validated predictions (*PRE*); and the precision improvement rate (*PIR*). Figure 4.9 plots user effort and precision improvement and, on the secondary Y-axis, the values of the above indicators. Overall, the indicators are aligned with the convergence of the validation process. For example, using the *URR* indicator, validation can be stopped at an *URR* value of 20%. Then, at 40% of user effort, large relative improvements of precision ($> 80\%$) have materialised already.

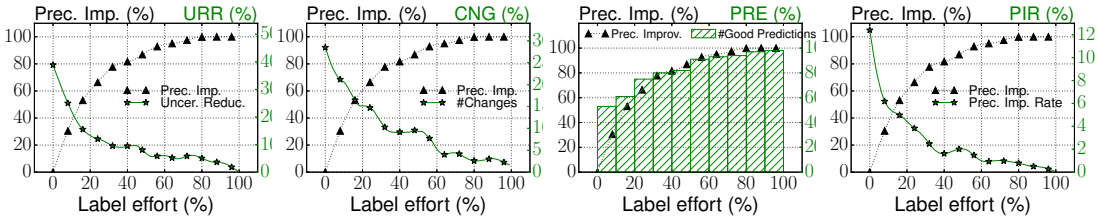


Figure 4.9: Effectiveness of early termination criteria

4.8.7 Benefits of Batch Validation

Next, we evaluate the benefits of selecting the top-k claims for validation. Here, high values of k lead to larger savings of user set-up costs. Yet, increasing k also implies less accurate estimation of potential benefit, due to our greedy algorithm (Section 4.6.2). To explore this trade-off, we capture the costs saved (CS) as a function of k: $CS(k) = 1 - 1/k^{\alpha}$, where α is the rail factor to control the increased cost of validating sets of claims. The chosen function form enables us to capture both linear and non-linear cost models in human validation practice.

Static batch size. When conducting validation with batching, the obtained precision will be lower, since inference is conducted only once the input for the whole batch has been incorporated. We measure this effect by the precision degradation, the relative difference in precision between the validation processes without batching and with batches of size k, varied between one and 20. Figure 4.10 plots precision degradation (%) relative to the cost saving (%) using batching under cost models with $\alpha = 0.25, 0.5, 1$. As expected, larger batches lead to lower precision, but increased cost savings. Medium-sized batches ($k = 5, 10$) appear to be beneficial, as they yield potentially large cost savings with a graceful reduction in precision.

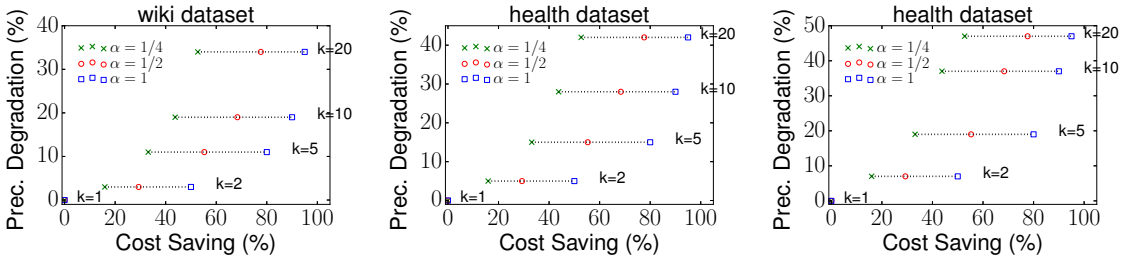


Figure 4.10: Effects of static batch size

Dynamic batch Size. We further explore a dynamic selection of the batch size k, with the goal to maximizes cost savings and precision. We consider different precision thresholds (0.8,0.9) and count the validated claims after each user interaction needed to reach that threshold. For a cost model with $\alpha = 2/3$, Figure 4.11 shows box plots of the user effort (%) relative the cost savings (%). Observing the same trade-off as in Figure 4.10, the specific results suggest how to choose k dynamically: Initially, a small k shall be used, which is increased once a sufficient amount of claims has been validated.

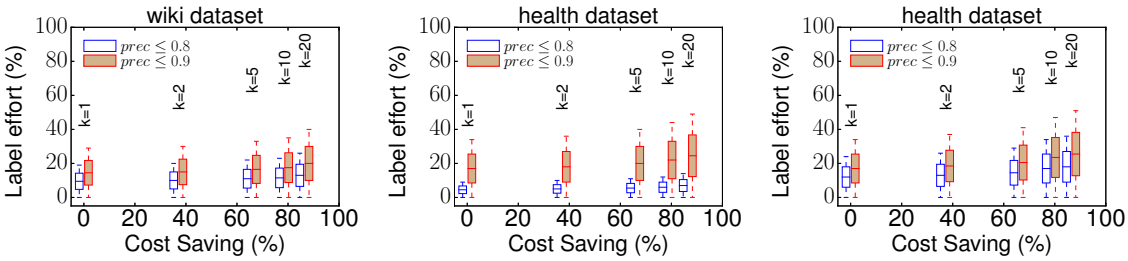


Figure 4.11: Effects of dynamic batch size

4.8.8 Streaming Fact Checking

Update time. We measure the response time during one iteration of Algorithm 5.1, i.e., the update time of the model when a new claim arrives. We run the update process from 0% to 100% of claims in the order of their posting time, for each dataset. The average update time for the *wiki*, *health*, and *snopes* datasets are 0.34s, 0.61s, and 1.22s respectively. As such, the response times turn out to be similar to those of Algorithm 4.1, as implied by Proposition 4 and Proposition 5.

Preservation of validation sequence. As explained in Section 4.7, the algorithms for streaming fact checking (Algorithm 5.1) and validation (Algorithm 4.1) run in parallel and update the model parameters. This leads to the question of how to interleave both algorithms: Validating claims early may not be beneficial as later arriving claims help in user guidance. To answer this question, we compare the validation sequences between the offline setting and the streaming setting as follows. We run the streaming algorithm from 0% to 100% of claims in the order of their posting time, and periodically invoke the validation process, where a claim is selected from the existing claims for validation (*hybrid* strategy, current model parameters provided by the streaming algorithm). We record the validation sequence and compare it with the offline setting using Kendall's τ_β rank correlation coefficient [Agr10]. It ranges from -1 (reverse order) to 1 (same order), quantifying the similarity of the ranking in two validation sequences.

Table 4.2: Preservation of validation sequence (Kendall's τ_β)

Dataset	validation period			
	5%	10%	20%	30%
wiki	0.23	0.46	0.78	0.84
health	0.19	0.42	0.71	0.78
snopes	0.12	0.38	0.59	0.67

Table 4.2 presents the result when varying the validation period from 5% to 30% (e.g., validation is invoked after every 5% of new claims arrive). Increasing this period,

the validation sequence of streaming fact checking becomes more similar to the static setting, as more information is accumulated in each period.

4.8.9 Real-world Deployment

Finally, we investigate practical issues when deploying our validation framework. A challenge for such an evaluation is that it is difficult to find experts that are knowledgeable in the domains covered by the annotated datasets. Therefore, we consider a setting that features supporting information for the validation. To derive this supporting information, we queried the Google search engine with the text of each claim and extracted the first ten search results as a list of documents. The list of documents is then shuffled for each validation task to avoid biases by the search engine or the user. Due to budget constraints, we selected 50 claims randomly for each dataset. Then, we considered two different types of users:

Experts (E): We implemented a validation interface for expert users, which records the time spent on validation and computes the average accuracy by comparing the answers with the ground truth. We asked three senior computer scientists to complete the validation tasks, with the option to pause between handling different claims.

Crowd workers (C): While it is not the primary use case for our work, crowdsourcing enables scaling of manual validation tasks with the risk of lower result quality due to different levels of worker reliability [HTTA13]. We used FigureEight [Eig18] and its web templates to deploy our validation tasks. We prepared a budget of 1500 HITs (Human Intelligence Tasks) in total with a financial incentive of 0.1$/HIT. We recorded the time spent on validation and computed the consensus of the answers among crowd workers using existing algorithms that include an evaluation of worker reliability [HTTA13]. The consensus answer is then compared to the ground truth.

Table 4.3: Avg. time and accuracy of experts and crowd workers

Dataset	Expert Time	Crowd Time	Expert Accuracy	Crowd Accuracy
wiki	268s	186s	0.99	0.88
health	1579s	561s	0.94	0.83
snopes	559s	336s	0.96	0.85

Table 4.3 summarises the obtained results. Experts validate claims more accurately than crowd workers, but take more time to complete. Moreover, the system also reports that the experts do not validate all the claims in one shoot; the validation process spanned 3-7 days. Note that in our setting, experts and crowd workers already had supporting information in place. Without it, they would have to retrieve such information on their own, which may further increase the validation time. The trade-offs illustrated in Table 4.3, however, point to the potential benefit of combining the input of experts and crowd workers to achieve efficient, yet accurate fact checking.

4.9 Summary

In this chapter, we proposed an approach to overcome the limitations of existing methods for automatic and manual fact checking. We introduced an iterative validation process, which, based on a probabilistic model, selects claims for which validation is most beneficial, infers the implications of user input, and enables grounding of the credibility values of claims at any time. We further proposed methods for early termination of validation,

efficient batching strategies, and a streaming version of our framework. Our experiments showed that our approach outperforms respective baseline methods, saving up to a half of user effort when striving for 90% precision.

Chapter 5

Misinformation Visualisation: The Case of Minimal-Regret Data Stream Retaining

Retaining Data from Streams of Social Platforms with Minimal Regret

IJCAI 2017

Today's social platforms, such as Twitter and Facebook, continuously generate massive volumes of data. The resulting data streams exceed any reasonable limit for permanent storage, especially since data is often redundant, overlapping, sparse, and generally of low value. This calls for means to retain solely a small fraction of the data in an online manner. In this chapter, we propose techniques to effectively decide which data to retain, such that the induced loss of information, the regret of neglecting certain data, is minimized. These techniques enable not only efficient processing of massive streaming data, but are also adaptive and address the dynamic nature of social media. Experiments on large-scale real-world datasets illustrate the feasibility of our approach in terms of both, runtime and information quality.

5.1 Introduction

Current social platforms such as Twitter, Facebook, and Yelp, produce data streams with an unprecedented rate. For example, about half a billion tweets are generated every day [MK10]. To make sense of these streams, one can typically only retain a small fraction of the data for analysis, due to storage limits and the cognitive load induced by the sheer data volume.

Against this background, traditional methods for the analysis of social platforms perform data summarization, e.g., based on relevance detection or measures of information diversity [CTY+16, ZRH+16]. However, these approaches are inherently limited to a static setting: The data is crawled and stored, before the top-k most important data items are selected as a data summary. Even if feasible, such an approach incurs high storage cost and does not avoid the problem of retaining only a fraction of the data once its volume exceeds a storage limit.

In this work, we consider the natural setting of social platforms, where data is dynamic and available as a stream. Then, retaining of data becomes more challenging

compared to one-off summarization, as data selection has to be repeated every time new data arrives. Instead of considering the whole historical data, summarization now works on the retained data (i.e., a previous summary) and the new data. This further degrades the informativeness of the original data since the summary of the retained data already induces some loss of information. Specifically, the lack of historical data leads to a biased assessment of data importance: Data that was considered unimportant and thus discarded in the past may retrospectively turn out to be important.

To minimize the regret of discarding important data, two requirements have to be met. First, a compact data sketch needs to be maintained, in addition to the actual data summary, to capture the long-term history of data and enable a precise assessment of data utility over time. Yet, for data stemming from social platforms, this sketch needs to be adaptive to changes in the data stream. Second, a protocol needs to be specified to decide which data items to retain and to discard, such that the total regret in data utility is minimal. To cope with the data stream volume and velocity of social platforms, this protocol needs to be very efficient.

Our approach. In this chapter, we tackle these requirements and propose a novel statistical model, which does not only capture the traditional context of social data (importance of topics, user influence, information diffusion) [MWDNM14b, ZRH+16], but also embeds the dynamics of this context over time. For example, topics are not considered to be static. They may emerge or disappear over time and relate to recurring events. Striving for online processing of streaming data, we develop a scalable learning mechanism to quickly update the model with new data. We further show how the statistical model is used to define a utility function to assess the representativeness of a data summary. Minimizing the regret of discarding data then becomes the problem of minimizing the difference between the utility of the retained data and the utility of whole historical data. Finally, we present a progressive algorithm to select which data to retain, with guarantees on the induced regret factor. This algorithm scales linearly in time and space, solely in the size of the data summary (not the whole data stream).

Our contributions and the chapter structure are summarized as follows. After outlining the retaining problem with minimal regret (Section 5.2), we present (i) a statistical model to sketch data properties; (ii) a utility function to assess data representativeness (Section 5.3); and (iii) a progressive algorithm to solve the retaining problem (Section 5.4). Using diverse datasets derived from Twitter, we demonstrate improvements of five orders of magnitude in efficiency and up to 42% in information quality of our approach over state-of-the-art baselines (Section 5.5). We then conclude the chapter (Section 5.6).

5.2 Problem Statement

We model the stream of data stemming from a social platform by an infinite set of textual data items $E = \{e_1, e_2, \ldots\}$. The items are totally ordered based on their occurrence time, denoted by the subscript. By $E_t = \{e_1, \ldots, e_t\}$, we denote the set of items until time t. Acknowledging that not all items from E can be stored permanently, a representative subset of E of size k shall be retained. Here, the parameter k depends on the application context and typically reflects the storage limit. We further postulate a non-negative function $f : 2^E \to \mathbb{R}_{\geq 0}$ to quantify the utility of a set of items $S \subseteq E$, capturing how well S represents E according to some objective. Given that E is continuously extended with new data items, the *retaining problem with minimal regret* is to select k items, such that the regret ratio—the normalized difference between the

utility of the retained items and the utility of the whole data stream—is minimal.

Problem 3 (Retaining Problem with Minimal Regret)**.** *Given a data stream E_t until time t, a current set of retained items $S_t \subseteq E_t$, a window of new data items $W = \{e_{t+1}, \ldots, e_{t+|W|}\}$, the problem is to construct a new set of retained items S_{t+w}, such that:*

$$S_{t+w} = \underset{S \subseteq (S_t \cup W),\ |S|=k}{\arg\min} \frac{f(E_t \cup W) - f(S)}{f(E_t \cup W)}. \tag{5.1}$$

The problem setting is illustrated in Figure 5.1. Upon the arrival of a window of new items, the set of retained items is updated. This is done by selecting items from the old set of retained items and from the window.

The figure further illustrates why the retaining problem with minimal regret cannot be addressed by applying traditional data summarization each time a window of new data items arrives. Traditional summarization would consider solely the current set of retained items and the items of the new window. Yet, the data stream history in terms of items discarded in the past would be neglected. Consequently, when constructing a new set of retained items, the utility of possible candidate sets cannot be assessed accurately.

As illustrated in Figure 5.1, therefore, a concise sketch of historical data needs to be maintained. It captures essential properties of data items that have been discarded in the past, thereby enabling an accurate assessment of the regret ratio of a potential set of retained items. To realise such a sketch, Section 5.3 presents a statistical model and also shows how to assess the representativeness of an item set using an utility function. In Section 5.4, we then present a progressive algorithm to solve the retaining problem with minimal regret under this model.

5.3 Model and Approach

Below, we first propose a statistical model to sketch the historical data of a stream, before turning to the question of how to assess the utility of a set of retained data items. Finally, we discuss a simple strategy to solve the retaining problem.

5.3.1 A statistical model for social data

For textual data items that originate from social platforms, topics are a fundamental concept to understand the co-occurrence relations of words [CSG09, CBHC13]. We thus

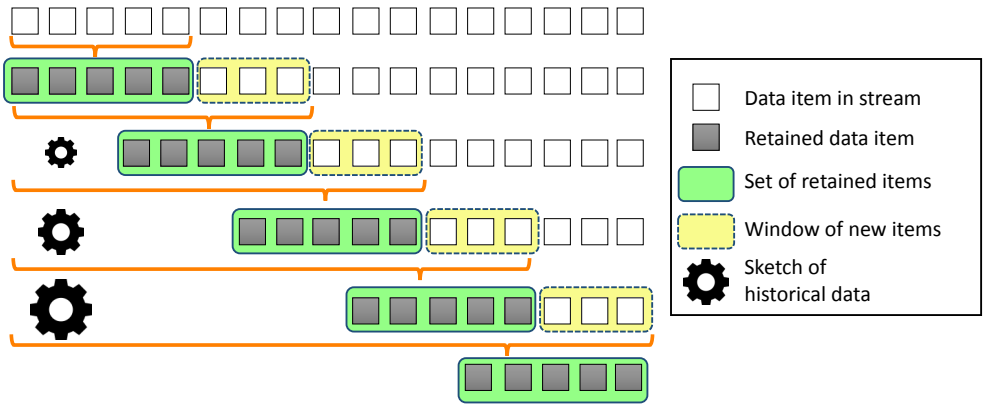


Figure 5.1: Illustration of the retaining problem ($k = 5$, $|W| = 3$).

capture information on topics as a statistical means to select representative data items from important clusters of words. However, the dynamic nature of streaming data from social platforms prevents us from knowing a specific distribution of topics in some future state. Rather, we face the following phenomena:

- *Emergent topics*: In a streaming setting, new topics may emerge over time. Hence, topic modelling techniques such as LDA [BNJ03] or pLSI [Hof99] that fix a predefined number of topics are not applicable. A small number of topics may lead to information loss, as different words might end up in the same cluster, whereas a large number of topics may imply sparse clusters, destroying data regularities.
- *Emergent vocabulary*: An evolving collection of topics implies that the vocabulary of words changes over time. Words may only be invented at a specific time [ZBG13] (e.g., 'brexit' during the events in the UK in June 2016). A fixed vocabulary as in traditional topic models [BNJ03, GT04] does not capture these dynamics and tends to be inefficient due to unused and redundant words.
- *Recurring topics:* Traditional data summarization typically spans a short period of time [CP11, ZRH+16], due to data storage limits. In contrast, when processing data streams of social platforms, a long history of data items is considered, so that topics will recur over time [HD14] (e.g., the topic of 'football' shows seasonal patterns). Such effects influence the decision of which data items to retain.

Against this background, we propose a non-parametric [HWC13] probabilistic model to sketch the properties of past data items. It features a potentially unbounded number of topics and words that are learned from textual data items over time. It also considers recurrent topics by means of temporal cluster variables, for which the time granularity can be customised.

Formally, a textual data item e is modelled as a multiset of words $\{v_1, \ldots, v_{|e|}\}$, where v_i is a word from a dynamic vocabulary V. A multiset of words further defines a semantic topic z. We model the dynamics of words per topic by means of a Dirichlet process to generate the word distribution ϕ_z and the vocabulary ρ_z of a topic z. Further, at time t, a topic distribution θ is used to generate the topics of a new data item e_t. To ensure that the temporal aspect of evolving topics is reflected in the generation, we use a hierarchical Dirichlet process to establish the link between data items in terms of topics. This yields a a concise and consistent set of topics rather than a sparse and unnecessarily large one. Finally, we model the recurring topics by means of temporal clusters, whose number is unbounded in general. To this end, a Chinese restaurant process [BGJ10] τ is used to non-parametrically generate a temporal cluster label c_t for each data item e_t. The model is summarised in Figure 5.2.

Generative process. The generative process for our model is defined as follows. For each item $e_t \in E$ at time t:

(1) Generate the lengths (number of words) of e_t from a Poisson distribution:
$N_e \sim Poiss(\epsilon)$

(2) Generate the topic of e_t from a categorical distribution:
$c_t \sim Cat(\tau)$

(3) Generate the temporal cluster label of e_t from a multinomial distribution:
$h_t | c_t, \pi \sim Mult(\pi_{c_t})$
where the value of h_t depends on the granularity according to which the topic distribution is captured for the data stream (e.g., h_t has a value of 31 to model topics per day of a month).

(4) For each of the $i = 1 \ldots N_e$ word indices of e_t:
a. Generate a topic from a multinomial distribution:

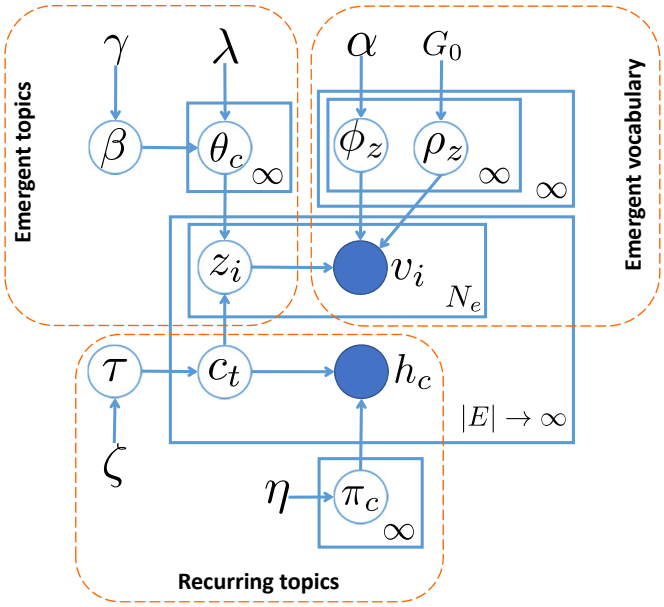


Figure 5.2: Model to sketch historical data. Shaded/blank circles are observed/latent variables, non-circles are model parameters.

$z_i|c_t, \theta \sim p(z|t) \triangleq Mult(\theta_{c_t})$

b. Generate a word v_i from z_i via a multinomial distribution:
$v_i|z_i, \phi, \rho \sim p(v|z) \triangleq Mult_{\rho_{z_i}}(\phi_{z_i})$
where ρ_z is a vocabulary generated from an n-gram model and ϕ_z is the probability of selecting ρ_z for word v_i [ZBG13].

Model parameters. The sketch of historical data is designed by parametrising the model as follows. The sketch is denoted by $\Theta = (\alpha, G_0, \lambda, \gamma, \zeta, \eta, \epsilon)$, a vector of parameters for:

- The Dirichlet process [Fer73] to generate ρ and ϕ:
  $\rho_{zk}, \phi_{zk} \sim DP(\alpha, G_0)$, for $t = 1, 2, \ldots$ and $k = 1, 2, \ldots$.
- The hierarchical Dirichlet process [TJBB06] to generate θ:
  $\beta \sim GEM(\gamma), \theta_c \sim DP(\lambda, \beta)$, for $c = 1, 2 \ldots$
- The Chinese restaurant process [BGJ10] to generate τ:
  $\tau \sim CRP(\zeta)$
- The Dirichlet distribution [Fer73] to generate π:
  $\pi_c \sim Dir(\eta)$, for $c = 1, 2, \ldots$

This parametrisation of the model yields a compact, light-weight sketch of historical data. Since it contains solely single-variable parameters, it has constant-space requirements. In Section 5.4, we will show how to update the model parameters based on a data stream by means of an incremental inference mechanism.

5.3.2 Utility of retained data

To judge how well a set of retained data items stemming from social platforms represents a whole data stream, we argue that the following aspects shall be considered: First, *semantic* information such as topics and word frequencies has to be incorporated, see [BNJ03, MPLC13, ZRH+16]. Second, the importance of data from social platforms is influenced by the *social context* (e.g., the authors of a textual statement) and its *fresh-*

ness, as the interestingness of social data degrades over time. Given the above sketch of historical data $\Theta^{(t)}$ at time t, these incorporate these aspects in the assessment of data utility as follows.

On semantic information. We first consider the probability $p(v, e)$ for observing a word w in a data item e at time t. It is defined based on the evolution of topics over time as

$$p(v, e) = \mathbb{E}_{p(z|\Theta^{(t)})} p(v|z)$$

where $p(v|z)$ represents the probability distribution of words given a topic, and $p(z|\Theta^{(t)})$ represents the probability distribution of topics at time t. The latter is derived from sketch Θ by the aforementioned generative distributions.

Based thereon, the probability of an item e being semantically important is defined as

$$p(e|\Theta^{(t)}) = \prod_{v \in e} p(v, e|\Theta^{(t)}) = \prod_{v \in V^{(t)}} p(v, e)^{n(v,e)}$$

where $V^{(t)}$ is the vocabulary at time t (maintained based on $\Theta^{(t)}$ in constant space [ZBG13]) and $n(v, e)$ denotes the frequency of a word $v \in V^{(t)}$ in item e.

On freshness and social context. To model that the interestingness of data degrades over time, we define a monotonic decreasing function $g(t)$. Specifically, the decay in interestingness of past data is described by an exponential form:

$$g(e) = \exp^{-\lambda(t - t(e))}$$

where λ is the decay rate and is set to 0.5 (maximal entropy principle), t is the current time and $t(e)$ is the time of e. Following [MWDNM14b, ZRH+16], we further associate each item e with a vector of social features $(h_1(e), \ldots, h_m(e))$. Then, the aggregation of these features, denoted by $h(e)$, describes the social context of a data item.

A utility measure. In social data, topics with highly frequent words may dominate other topics. To avoid such vocabulary bias, we define the utility of a set of items S as the log-likelihood over its items based on the information entropy. This measure of utility incorporates the above notions of semantic information, freshness, and social context:

$$f(S) = \sum_{e \in S} \sum_{v \in V} n(v, e) p(v, e) \log \frac{1}{p(v, e)} g(e) h(e) \tag{5.2}$$

Using this formulation, an algorithm to select data items will prefer sets with high entropy, i.e., sets with items that cover diverse topics and preserve the evolving topic distribution. As proven below, the above measure is monotonic (selecting more items increases utility) and submodular (marginal gains by selecting more items start to diminish due to saturation of the utility objective).

Proposition 1. *$f(.)$ is a monotonic function.*

Proof (Sketch). Let E be a sequence of items, $S \subset E$ is a selection, and $e \in E \setminus S$ is from a set of non-selected tweets. Then it holds that: $f(S \cup \{e\}) \geq f(S)$. Indeed, we denote all the words that occurs in e but not in the selection S as the set of words $e \setminus V_S$. Then we have

$$f(S \cup \{e\}) - f(S) = \sum_{v \in e \setminus V_S} n(v, e) p(v, e) \log \frac{1}{p(v, e)}$$

, which is non-negative. □

Proposition 2. *$f(.)$ is a submodular function.*

Proof (Sketch). Let E be a sequence of items, S a selection, and $e, e' \in E \setminus S$ from a set of non-selected tweets. Then it holds that: $f(S \cup \{e\}) - f(S) \geq f(S \cup \{e, e'\}) - f(S \cup \{e'\})$. Similar to the proof of monotonicity, we expand the inequality to:

$$\sum_{v \in e \setminus V_S} n(v,e)p(v,e) \log \frac{1}{p(v,e)} \geq \sum_{v \in e \setminus V_{S \cup \{e'\}}} n(v,e)p(v,e) \log \frac{1}{p(v,e)}$$

, which is equivalent to $\sum_{v \in e'} n(v,e)p(v,e) \log \frac{1}{p(v,e)} \geq 0$. The equality happens if and only if $e \cap e' = \emptyset$. □

5.3.3 A simple retaining algorithm

A straight-forward approach to solve Problem 3 under the above model applies traditional data summarization [ZRH+16] on the retained items and the content of a new window. Then, the new set of retained items is selected as:

$$\max_{S \subseteq S_t \cup W, |S|=k} f(S) \tag{5.3}$$

which is equivalent to Equation 5.1 since the value of $f(E_{t+w})$ is constant in terms of selecting any S. However, this problem is known to be NP-complete [ZRH+16, NW81, HTWA15c]

Due to the computational complexity, greedy approximation algorithms (inspired by the knapsack problem) are commonly employed. They start with the empty set $S^{(0)} = \emptyset$, and at each iteration i over the current data, choose an item $e \in S_t \cup W$ maximizing the utility, i.e.,

$$S^{(i)} = S^{(i-1)} \cup \arg\max_{e \in S_t \cup W} f(S^{(i-1)} \cup e) - f(S^{(i-1)}) \tag{5.4}$$

For a monotonic and submodular function (as our utility function defined above), this greedy algorithm yields a $(1 - 1/e) \approx 0.63$ approximation [NW81]. However, this algorithm has an update time complexity of $\mathcal{O}(k(k+|W|))$, which is undesirable for streaming applications. Also, the update process needs to be repeated every time new data arrives.

5.4 A Progressive Retaining Algorithm

Given the above results on hardness of the retaining problem and the time complexity of a simple greedy algorithm, we now present a progressive algorithm. It is tailored to the stream processing setting and shows linear time and space complexity. The algorithm comprises of (i) an incremental inference mechanism to update the sketch of historical data; and (ii) a mechanism to select a new set of retained items.

5.4.1 Updating the data sketch

To update the parameters Θ of our model, we realise an online learning mechanism. When data arrives, we compute the observed variables and propagate back the information to the model parameters. Once the parameters have been updated, the conditional and marginal probabilities for the utility function are computed following the generative process.

Many online learning techniques have been developed based on Markov chain Monte Carlo sampling (e.g., incremental Gibbs sampling [CSG09]). However, these techniques

either reduce model complexity (loosing the guarantee to converge for the complete model) or have a space complexity that is linear in the number of items to analyse [HBB10]. To overcome these issues, we rely on stochastic variational inference [HBWP13]. Here, the idea is to minimize the evidence lower bound (ELBO) of the expected difference between the observed distribution and the latent distribution, defined as:

$$L(Z) = \mathbb{E}_{q(Z)}[ln\frac{p(Z,E)}{q(Z)}]$$

where Z is the set of all latent variables in the model (except model parameters and observed variables); and $q(Z)$ is an approximate distribution of $p(Z|E)$, which can be factorised over the distributions of model parameters in the generative process. Then, we apply stochastic optimisation to the ELBO function over a data stream, which basically updates the new parameter values from the previous ones following the direction of the ELBO gradient of new data with a fixed step size. The more data is received, the more the model parameters will converge to minimize the ELBO function.

Instead of single-item update, our approach considers multiple observations per update to reduce noise [HBWP13], which also aligns with window-based processing to avoid order distortion in data streaming settings. Receiving data as a series of windows W_b, $b = 1, 2, \ldots$, of items, we proceed as follows:

1. We update the local parameters of the variational distribution of the word-topic variable z and the topic-cluster variable c. This requires us to maintain additional M_z local parameters for possible values of z, and M_c local parameters for possible values of c. Here, M_c, M_z are 'prior beliefs' on the maximum number of topics and recurring topic clusters. Yet, their effects are marginal, as the updates are dominated by the observed information, so that they can be safely set to large constant values (e.g., 1000).
2. We compute the natural gradients using previous parameter values $\Theta^{(t-1)}$ and the above local parameters of the ELBO function decomposed over each item $e \in E_b$ [HBB10]. Formally, we obtain $\nabla_{\Theta^{(t-1)}} L_e$ as a vector of gradients for each parameter in $\Theta^{(t-1)}$.
3. The new values for model parameters are computed from their previous value:

$$\Theta^{(t)} = \Theta^{(t-1)} + w_b \frac{1}{|W_b|} \sum_{e \in W_b} \nabla_{\Theta^{(t-1)}} L_e \tag{5.5}$$

   where w_b is the learning rate to control the learning quality and convergence of the inference. w_b is often modelled as a power function of b with a forgetting rate r [HBWP13]. Setting $r \in (0.5, 1]$ guarantees convergence [HBWP13], while larger values often lead to higher learning quality and faster convergence (but not monotonically).

Note that windows of large size reduce the number of updates, but may lead to a poor estimation of model parameters. To further improve scalability, updating of model parameters can be parallelised by exploiting the conditional independence property. When the global variables (i.e., the most outer parameters in the model) are given, the updates to local variables (i.e., inner parameters) become independent and can thus be computed concurrently. Also, the computation of semantic information, decay in interestingness, and social features per data item, see Section 5.3.2, is independent once the model parameters are updated and thus can be parallelised.

5.4.2 Retaining data items

We now turn to the selection of data items to retain. In essence, at each step, a new set of items is selected as the one with the highest utility among all candidate sets. The candidate sets are created by swapping at most one new item with a retained item, if the utility increases. However, the arrival of a new item may change the model parameters, influences the decay in interestingness of old items, and may introduce new social features. Hence, the utility of the retained set of items has to be updated, before the swapping procedure is started.

Retaining algorithm. Our progressive retaining algorithm is formalised in Algorithm 5.1. We illustrate the algorithm with a window size of $|W| = 10$ (line 6). The algorithm starts with the empty set $S_0 = \emptyset$. As long as no more than k elements $e_1, \ldots, e_t$ have arrived, all of them are kept, i.e., $S_t = S_{t-1} \cup \{e_t\}$, for $t \leq k$. For each new item e_t, where $t > k$, we update the utility value of $f(S_{t-1})$ and compute the semantic importance $p(e_t)$, the decay in interestingness $g(e_t)$ and the social features $h(e_t)$. Then, we check whether swapping this item and an item in S_{t-1} will increase the utility value. If so, the one that maximizes the utility is selected for swapping.

Theorem 5.4.1. *Algorithm 5.1 does a single pass over data stream, uses $\mathcal{O}(k)$ memory, and has $\mathcal{O}(k)$ update time per item.*

Proof (Sketch). The proof is straightforward from the algorithm. The loops in line 2 and line 4 pass over the data stream only once. We need to maintain a frequency matrix and a probability matrix for $n(v, e)$ and $p(v, e)$ for all $v \in V$ and $e \in S_t$ where $|S_t| = k$. Other maintenance of $g(e)$ and $p(e)$ take only $\mathcal{O}(k)$ memory. Considering the number of model parameters as constant yields the required memory as $\mathcal{O}(k)$ and the update time per each loop in line 14 as $\mathcal{O}(k|W|)$. Since $|W|$ is often small ($|W| \ll k$) and can be considered as constant, the update time complexity becomes $\mathcal{O}(k)$. □

Correctness of Algorithm 5.1 is established as follows: First, the decay in interestingness is a monotonic decreasing function. Thus, an optimal selection remains optimal after the utility has been updated. Also, even if a new item does not increase utility in terms of entropy and social features, the algorithm still swaps it with an old item to preserve the freshness of the retained set of items. Second, adding a new item increases the entropy of the topic distribution. Thus, while the algorithm favours new items, it still preserves topics by ensuring an even distribution of the selected items across all topics (old and new). Moreover, an optimal selection remains optimal as the entropy increases for all old items.

Incremental utility computation. The above complexity result assumes that the computation of utility (line 14), when swapping data items, is done in constant time. This indeed holds true, since utility can be computed incrementally, i.e., $f(S_i)$ is derived from $f(S_{i-1})$ in constant time as follows:

$$f(S_{i-1} \setminus \{e\} \cup \{e'\}) = f(S_{i-1}) - \sum_{v \in V} n(v, e) p(v, e) \log \frac{1}{p(v, e)} g(e) h(e) + \sum_{v \in V} n(v, e') p(v, e') \log \frac{1}{p(v, e')} g(e') h(e').$$

Here, values $p(.,.)$, $n(.,.)$, $g(.)$, $h(.)$ have been computed already in the previous steps of the algorithm (lines 9 to 13).

Algorithm 5.1: A Progressive Retaining Algorithm

input : An infinite sequence E of data items
output: A selected set S_t of size k of data items at any time t

$S_0 = \emptyset$;
for $t = 1$ *to* k **do** $S_t = S_{t-1} \cup \{e_t\}$;
$W = \emptyset$; ▷ Sliding Window
for $t = k + 1$ *to* $|E|$ **do**
  $W = W \cup \{e_t\}$;
  if $|W| < 10$ *and* $t < n$ **then**
    $S_t = S_{t-1}$;
    continue;
  `// Incremental learning of model parameters`
  Compute $\Theta^{(t)}$ from $\Theta^{(t-1)}$ and W;
  Update $p(v, e)$ and $n(v, e)$ by new parameter $\Theta^{(t)}$, $\forall v \in V, e \in S_{t-1}$;
  Update $g(e)$ $\forall e \in S_{t-1}$;
  Update $f(S_{t-1})$;
  `// Prepare computation of utility of new items`
  Compute $p(v, e')$, $n(v, e')$, $g(e')$, $h(e')$ for all $e' \in W$ and $v \in V$;
  `// Find swapping pair`
  $e^*, e_b = \arg\max_{e \in S_{t-1}, e' \in W} f(S_{t-1} \setminus \{e\} \cup \{e'\})$;
  if $f(S_{t-1} \setminus \{e^*\} \cup \{e_b\}) \geq f(S_{t-1})$ **then**
    $S_t = S_{t-1} \setminus \{e^*\} \cup \{e_b\}$;
  else $S_t = S_{t-1}$;
  $W = \emptyset$; ▷ Reset for new window

5.5 Empirical Evaluation

Below, we first elaborate on the experimental setup, before we analyse our method's efficiency and effectiveness.

5.5.1 Experimental Setup

Datasets. We extracted datasets using the Twitter Streaming API [ZRH+16]. Over a year, we considered five different domains (climate change, vaccination, processed food, genetically modified organism, general public) and randomly selected 1 million English tweets per domain. Furthermore, a total of five important social features (e.g., user influence, retweet score, and affective language) had been extracted for each data item using existing frameworks [ZRH+16].

Baselines. We compare our approach with several baselines:

- *Traditional summarization:* a state-of-the-art summarization technique [ZRH+16] for social data.
- *Greedy*: the simple greedy algorithm (see Section 5.3.3) to select the items to retain.
- *Offline learning*: an iterative algorithm to compute model parameters using deterministic variational inference [JGJS99, BJ+06], which requires a full pass of the data in each iteration.
- *Static:* our retaining algorithm tailored to the traditional, static setting of data summarization: *offline learning* to compute the sketch of historical data and the *greedy* algorithm to select the items to retain.

Environment. All results have been obtained on an Intel i7 3.8GHz system (4 cores, 16GB RAM). Following [HBWP13, HBB10], we vary the forget rate in $(0.5, 1]$, choose a stable window size $= 10$ and report average values.

5.5.2 Efficiency

We evaluate the update time of our approach, when new data arrives. To assess the average time per window needed to update the model parameters, we compare our online learning algorithm with its offline version. The latter considers the whole data received so far when computing the parameters upon the arrival of a new window. Figure 5.3 illustrates the results averaged over all datasets, reporting the average update time until $100K$, $500K$, and $1M$ data items are received. Here, the update time of our progressive approach remains constant and small (< 0.01s), whereas the baseline yields a high and increasing runtime (up to 10^3s).

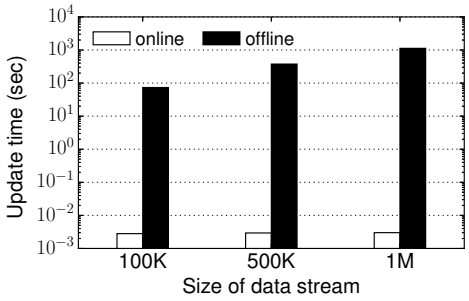


Figure 5.3: Model update

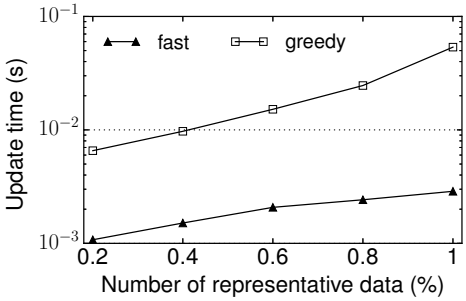
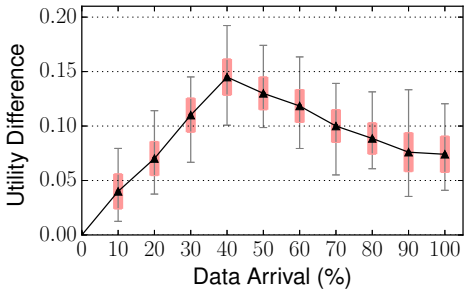


Figure 5.4: Retaining data

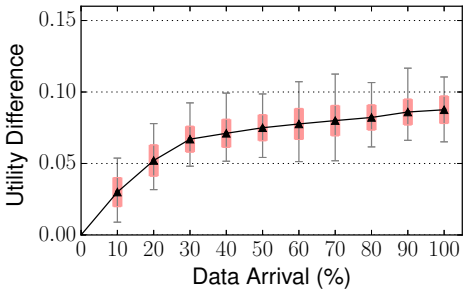
Focusing on how to select the items to retain, we compare the efficiency of our progressive algorithm (*fast*) with the *greedy* algorithm. We realise the online setting as above and vary the size of the set of retained items ($k = 0.2\%$ to 1%). Figure 5.4 depicts the update time of the algorithms, averaged over all datasets. Our progressive algorithm outperforms the *greedy* one. It also scales better to large data summaries.

5.5.3 Effectiveness

We compare the quality of model parameters, in terms of utility, obtained with our online learning algorithm and its offline version. To mitigate the randomness of the Twitter streaming API, we select 100K items $\tilde{E}$ from the original datasets and construct a set of retained items S as the $k = 1\%$ oldest items in $\tilde{E}$. We stream $\tilde{E}$ and learn model parameters online ($\Theta_{\tilde{E}}$). Offline learning considers all data received so far, the result being $\Theta'_{\tilde{E}}$. We then assess the relative difference in utility of S, computed with either method, i.e., $\frac{|f_{\Theta}(S) - f_{\Theta'}(S)|}{f_{\Theta'}(S)}$.



(a) Updating model parameters



(b) Retaining data items

Figure 5.5: Effectiveness relative to amount of processed data

The difference in utility relative to the amount of processed data is shown in Figure 5.5a (averaged over 100 runs of different $\tilde{E}$ and the five datasets). Due to the stochastic property of online learning (data is needed to converge in the model parameters), the utility increases initially. Also, the difference between the online and offline learning results is less than 15% in general, underlining the usefulness of our approach.

We further compare the utility of retained items selected by the greedy and our progressive algorithm. Similar to the above setting, we stream all data and use online learning to update the model parameters. Both algorithms start from the set of $k = 1\%$ old items and update it upon the arrival of a new window. We then measure the relative difference of the obtained sets of retained items at each step. The results in Figure 5.5b (averaged over 100 runs and the five datasets) show that the utility difference increases with the arrival of data. This is because the progressive algorithm accumulates some loss of information. However, the difference is small ($\leq 15\%$) and converges with more data. As such, data quality is not compromised too much.

Table 5.1: Overall utility ratio

k		climate	vacc	food	gmo	public
1%	static	0.84	0.83	0.84	0.81	0.86
	dynamic	0.70	0.74	0.73	0.74	0.76
	sum	0.72	0.70	0.69	0.71	0.71
.1%	static	0.74	0.76	0.73	0.72	0.75
	dynamic	0.69	0.67	0.68	0.65	0.68
	sum	0.51	0.53	0.48	0.51	0.52

Finally, we compare the overall utility ratio (utility of output over utility of whole data) of our retaining algorithm (*dynamic*) and two baselines: the *static* version of our algorithm and traditional summarization (*sum*). The utility ratios obtained after processing all data of the five datasets are shown in Table 5.1, for different sizes of the set of retained items ($k = 0.1\%$ and $k = 1\%$). While the *static* approach outperforms traditional summarization (*sum*), we need to acknowledge that both, *static* and *sum*, are inapplicable for streaming data. However, the results of the *dynamic* technique are relatively close to those of the *static* approach, highlighting its usefulness for online processing. Also, its decrease in utility for a smaller number of retained items (k) is less drastic compared to traditional summarization.

5.6 Summary

This chapter proposed a technique to retain a representative set of items from a stream of social data. That is, we acknowledge the online nature of data produced by social platforms, which prevents us from storing the complete data stream. This led the retaining problem with minimal regret, where a protocol decides which data to retain, such that the loss of utility is minimized. To address this problem, we proposed a light-weight, adaptive sketch of historical data and a progressive algorithms for the selection of data items. Experiments on large-scale real-world data showed that our approach is efficient (five orders of magnitude faster than the baseline) and effective (less than 15% reduction in the utility ratio).

Chapter 6

Conclusion

6.1 Summary of the Work

With the help of Web technologies, the penetration of social platforms in general and social media in particular into our lives have increased, enabling both ordinary users and professionals to consume news in a real-time fashion. While the proliferation of these information diffusion channels help modern society to overcome communication barriers, the unverified nature of social data gives birth to misinformation both intentionally and unintentionally, ranging from "honest" journalists' negligence, to large-scale orchestrated campaigns. Since malicious contents are generally crafted to resemble a genuine news offer, it is hard for the public to distinguish fake news from legitimate articles without proper alerts from algorithmic models. This creates direct threats to democratic political processes and social values including health, finance and scientific findings. Developing resources to understand and mitigate the impact of fake news is, therefore, of tremendous importance [VRA18].

Misinformation propagation on social platforms could largely affect our daily lives. Governments across the world are considering misinformation as a cyber threat to digital democracies. For example, the European Union and the Australia Cybersecurity Centre expressed concerns that social media users were over-exposed to extremist and terrorist content online [gov, Cen]. UK parliament is also formulating stricter regulations on the tech giants such as Facebook due to their recent failure against some fake news [BBC]. Researchers and media giants are concerned that misinformation on the Web can alter public opinions against verified scientific reports. A popular example is climate change denial [Gua], in which repeated targeted attacks from climate change sceptics through Facebook can make users highly susceptible to commercial bias.

Although many misinformation studies have become increasing popular [VRA18, CGL+18, ZAB+18, Zea18], debunking online misinformation is an endless battle due to the evolving traits of misinformation stories. More and more forms of falsehoods arise such as rumours, false claims, post-truths, clickbaits, and fake news. A complete misinformation debunking framework that incorporates both algorithmic models and human experts to fight misinformation attacks in a systematic and adaptive manner is still missing. Towards this goal, this thesis presented solutions to typical misinformation domains related to the cornerstones of such framework, including *Detection*, *Validation*, and *Visualisation*. While the first one will discuss how to develop automatic algorithms for online setting, the last two will show how to involve and support humans in the automatic processes.

Misinformation Detection – The Case of Rumour Detection. Chapter 3 proposed an approach for rumour detection that is grounded in anomalies in the data of a social platform. We presented methods to detect anomalies at both, the local and global level of a social graph. Local anomalies are based on historical data and similarities between entities. They serve as the basis to characterise global anomalies that represent rumours through the combination of information of different modalities. Our experiments showed that our method is effective and efficient, and detects rumours early and accurately. In particular, it outperforms several baseline methods, which are limited to a single modality of social data.

Misinformation Validation – The Case of Fact Checking. In Chapter 4, we proposed an approach to overcome the limitations of existing methods for automatic and manual fact checking. We introduced an iterative validation process, which, based on a probabilistic model, selects a claim for which validation is most beneficial, infers the implications of the user input on the fact database as a whole, and enables grounding of the credibility values of claims at any time. We further proposed different strategies to guide users and presented optimisations that increase the efficiency and robustness of the validation process. Our evaluation showed that our approach outperforms respective baselines methods significantly, saving up to 53% of user effort when striving for 90% result precision.

Misinformation Visualisation – The Case of Streaming Data. In Chapter 5, we study the problem of retaining representative data over social media streams. We motivate a "truly" online setting where data has to be chosen to retain or discard rather than being stored somewhere and processed later on in a streaming fashion. This setting is natural in the Big Data era, where the volume of data is much higher than user storage limit or data arrives in a very fast pace that does not allow much time to store the data. This leads to the minimal regret problem, where we need to design a protocol to retain the data such that the loss of utility is minimized. We show that such a problem can be solved by using a light-weight compact structure to incrementally adapt with the dynamics information of data and a fast progressive swapping mechanism to increase the utility when new data arrives. Experiments on real social data extracted from Twitter Streaming API show the efficiency of our approach (five orders of magnitude faster than the baseline), while proving an acceptable quality for user (less than 15% of utility regret).

6.2 Novelty and Limitations

This thesis is the first attempt towards building a continuous misinformation debunking framework by integrating algorithmic models and human validation in a closed loop process, where detection algorithms flag suspicious information waves for human attention, so that these waves are evaluated by human judges to make detection more accurate.

- A novel graph-based data representation model is developed to advance the granularity of existing techniques in handling misinformation signals. It leverages network structures of users, social links, and interactions as well as their dynamics to enable a more precise assessment of potential targets of misinformation waves.
- A novel non-parametric probabilistic model is developed to detect misinformation candidates based on inconsistencies that indicate an abnormal or unexpected evolution of the information waves. In particular, the model enhances the system

robustness to different types of misinformation by a Bayesian method that uses statistical tests rather than data distribution assumptions.

- A novel cost-effective human validation process on top of existing human validation platforms is developed to minimise validation costs and speed-up system response, paving the way for the development of resilience plans to mitigate the spread of misinformation.
- A novel data stream visualisation protocol is designed to address the issues of high velocity and high volume with massive social streams.

While our studies are limited to typical misinformation domains such as rumours and Web claims, the thesis outcomes are a series of computational algorithms, tools, pipelines, and guidelines extensible for several forms of misinformation such as fake news, malicious rumours, scams, and propaganda. While the thesis focuses on the particular effort minimisation problem of human validators, it will stimulate further research in unlocking the power of human computation, particularly expert knowledge, in the Big Data era. While important issues of misinformation detection, validation, and visualisation have been addressed in the thesis, further developments need to be done at the engineering level to enhance the public trust, to guard social media connected businesses from agenda-based attacks, and to protect policy makers from misinformation fuelled disruptions such as harmful protests, political echos, public opinion manipulation and altered election outcomes.

6.3 Future Directions

We recognize that the novel approaches described in this dissertation can be strengthened in a number of ways and open many opportunities for future work. We suggest the following research directions.

Misinformation Mitigation. While debunking misinformation is shown as a formidable challenge in this thesis, it is only a first step in preventing the spread of misinformation in social networks [Zea18]. From a practical perspective, the experimental findings in this thesis will help policy makers to identify where the issues are and formulate regulations accordingly. From a computational perspective, the developed graph-based detection method (Chapter 3), in particular understanding the echo chamber effect, will facilitate the developments of resilience plans, including:

- Monitoring and blocking in real-time the propagation paths of misinformation waves in a social network.
- Investigating and handling most influential users first to maximise the mitigation effect with minimal effort.
- Guiding and supporting social users participate in the debunking process by, e.g., letting them vote/rate a misinformation flag.
- Providing contextual information and evidences so that users can confidently counterargue a misinformation-related social post (e.g. by comments)
- Recommending reliable sources for users to explore about the topics they are most vulnerable because of their political biases and prior knowledge.

Transdisciplinary Framework. Our and existing misinformation studies are generally limited to a particular social network data and a misinformation domain (e.g. rumours, fake news) [VRA18, CGL+18]. Towards a robust and adaptive misinformation debunking framework, it is necessary to perform analyses across and combining different domains, platforms, traits, languages and topics of misinformation. Such holistic approach will enable the following advantages:

- Provide deep insights of social patterns (e.g. propagation, lexical, temporal) to understand the unique dynamics of each misinformation story.
- Overcome the sparsity issue when handling an individual misinformation domain alone by, e.g., combining all social posts, users, and social interactions in the same graph model. It also helps to identify social communities with multiple interests, facilitating more fine-grained credibility assessment and recommendations.
- Leverage the advances of machine learning, particularly deep learning, to implement effective prediction model using random forests and deep neural networks. In particular, the new representation learning enabled by deep learning architectures such as convolutional neural networks and recurrent neural networks helps to reduce the effort of heuristic-based feature engineering by automatically learning high-dimensional and discriminative features from large-scale and dense data.

Explainable Algorithmic Outputs. Existing algorithmic models stop at giving a classification result of a social unit is misinformation or not without further explanation. While they are the heart of Big Data Analytics today, giving them the right to decide the social outcomes could harm digital democracy. On the other hands, there is a gap of domain experts in understanding the behavior of generic algorithm models, making the integration of human knowledge and machine power difficult. To facilitate such problem, the following research directions could be explored:

- *What-if analysis:* algorithmic models are often sensitive to the parameter set-up and the discriminative features on which they are based. Predicting the outcomes of algorithmic models with synthesized settings could help to understand why they make that classification output and how the misinformation is fabricated. Techniques such as what-if analysis [NZW+18b] can help to analyse such sensitivity.
- *Evidence mining:* Computing explanations for an algorithmic decision is an established research direction [TM07]. In particular for social data, mining arguments to provide evidences for a given claim is an important step for users to understand the contexts, related scientific findings, and cross-checking information [HDT+17].
- *Partial misinformation:* Online information generated by social users has grown quickly in terms of volume and format. For example, Twitter doubled the character limit of tweets and allowed the attachment of multimedia such as images and videos. This leads to various types of information correctness, including *partially sound* information (e.g. only part of a social post contains false information) and *partially complete* information (e.g. a given social post does not report all aspects around an information story). To enable a more fine-grained insight and facilitate fairness, more research efforts have been spent for misinformation localization [CGL+18] and partial truth computation [HVT+18].

Bibliography

[ABC+18] John Akers, Gagan Bansal, Gabriel Cadamuro, Christine Chen, Quanze Chen, Lucy Lin, Phoebe Mulcaire, Rajalakshmi Nandakumar, Matthew Rockett, Lucy Simko, et al. Technology-enabled disinformation: Summary, lessons, and recommendations. *arXiv preprint arXiv:1812.09383*, 2018. 8

[ABR16] Nasser Alsaedi, Peter Burnap, and Omer Farooq Rana. Automatic summarization of real world events using twitter. In *ICWSM*, pages 511–514, 2016. 22

[ACMH03] Karl Aberer, Philippe Cudré-Mauroux, and Manfred Hauswirth. The chatty web: emergent semantics through gossiping. In *WWW*, pages 197–206, 2003. 11

[ACMO+04] Karl Aberer, Philippe Cudré-Mauroux, Aris M Ouksel, Tiziana Catarci, Mohand-Said Hacid, Arantza Illarramendi, Vipul Kashyap, Massimo Mecella, Eduardo Mena, Erich J Neuhold, et al. Emergent semantics principles and issues. In *DASFAA*, pages 25–38, 2004. 11

[AED99] Shlomo Argamon-Engelson and Ido Dagan. Committee-based sample selection for probabilistic classifiers. *JAIR*, pages 335–360, 1999. 20

[AGK10] Arvind Arasu, Michaela Götz, and Raghav Kaushik. On active learning of record matching packages. In *SIGMOD*, pages 783–794, 2010. 71

[AGMS13] Yael Amsterdamer, Yael Grossman, Tova Milo, and Pierre Senellart. Crowd mining. In *SIGMOD*, pages 241–252, 2013. 20

[Agr10] Alan Agresti. *Analysis of ordinal categorical data*. John Wiley & Sons, 2010. 76

[All17] Robert Allen. What happens online in 60 seconds? http://tiny.cc/m1my2y, 2017. 2

[ÁMLM13] Eduardo Álvarez-Miranda, Ivana Ljubić, and Petra Mutzel. The maximum weight connected subgraph problem. In *Facets of Combinatorial Optimization*, pages 245–270. Springer, 2013. 37

[ATH06] Duong Tuan Anh, Vo Hoang Tam, and Nguyen Quoc Viet Hung. Generating complete university course timetables by using local search methods. In *4th International Confernce on Computer Sciences: Research,*

Innovation and Vision for the Future, February 12-16, 2006, Ho Chi Minh City, Vietnam, pages 67–74, 2006. 12

[BBC] BBC. https://www.bbc.com/news/technology-47255380. 91

[BCC+16] Raphaël Bonaque, Tien Duc Cao, Bogdan Cautis, François Goasdoué, Javier Letelier, Ioana Manolescu, Oscar Mendoza, Swen Ribeiro, and Xavier Tannier. Mixed-instance querying: a lightweight integration architecture for data journalism. In *VLDB*, volume 9, pages 1513–1516, 2016. 11

[BCS+07] Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. Open information extraction from the web. In *IJCAI*, pages 2670–2676, 2007. 9

[BDF+13] Kalina Bontcheva, Leon Derczynski, Adam Funk, Mark Greenwood, Diana Maynard, and Niraj Aswani. Twitie: An open-source information extraction pipeline for microblog text. In *RANLP*, pages 83–90, 2013. 9

[BG94] Michael Buckland and Fredric Gey. The relationship between recall and precision. *Journal of the American society for information science*, 45(1):12–19, 1994. 28

[BGJ10] David M Blei, Thomas L Griffiths, and Michael I Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *JACM*, pages 7:1–7:30, 2010. 82, 83

[BH08] Philip A Bernstein and Laura M Haas. Information integration in the enterprise. *Communications of the ACM*, 51(9):72–79, 2008. 11

[BJ79] Robert H Berk and Douglas H Jones. Goodness-of-fit test statistics that dominate the kolmogorov statistics. *Probability theory and related fields*, pages 47–59, 1979. 37, 39

[BJ+06] David M Blei, Michael I Jordan, et al. Variational inference for dirichlet process mixtures. *Bayesian analysis*, pages 121–143, 2006. 88

[BJV15] Piyush Bansal, Somay Jain, and Vasudeva Varma. Towards semantic retrieval of hashtags in microblogs. In *WWW*, pages 7–8, 2015. 30, 44

[BLSR13] Floris Bex, John Lawrence, Mark Snaith, and Chris Reed. Implementing the argument web. *Communications of the ACM*, 56(10):66–73, 2013. 11

[BMKK14] Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. Streaming submodular maximization: Massive data summarization on the fly. In *KDD*, pages 671–680, 2014. 22

[BNJ03] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *JMLR*, pages 993–1022, 2003. 21, 82, 83

[Boe94] Lawrence E Boehm. The validity effect: A search for mediating variables. *Personality and Social Psychology Bulletin*, 20(3):285–293, 1994. 13

[BRLT+15] Senjuti Basu Roy, Ioanna Lykourentzou, Saravanan Thirumuruganathan, Sihem Amer-Yahia, and Gautam Das. Task assignment optimization in knowledge-intensive crowdsourcing. *VLDBJ*, 24(4):467–491, 2015. 68

[BWS+10] Steve Branson, Catherine Wah, Florian Schroff, Boris Babenko, Peter Welinder, Pietro Perona, and Serge Belongie. Visual recognition with humans in the loop. In *ECCV*, pages 438–451, 2010. 18

[BY14] Ricardo Baeza-Yates. Wisdom of crowds or wisdom of a few. *Web Engineering*, 573, 2014. 7

[CA13] Freddy Chong Tat Chua and Sitaram Asur. Automatic summarization of events from social media. In *ICWSM*, pages 81–90, 2013. 22

[Cam16] Erik Cambria. Affective computing and sentiment analysis. *IEEE Intelligent Systems*, 31(2):102–107, 2016. 9

[CBHC13] Youngchul Cha, Bin Bi, Chu-Cheng Hsieh, and Junghoo Cho. Incorporating popularity in topic models for social network analysis. In *SIGIR*, pages 223–232, 2013. 81

[CBK07] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Outlier detection: A survey. *ACM Computing Surveys*, 2007. 12

[CCAY16] Anand Inasu Chittilappilly, Lei Chen, and Sihem Amer-Yahia. A survey of general-purpose crowdsourcing techniques. *TKDE*, 28(9):2246–2266, 2016. 19

[CCWH18] Iti Chaturvedi, Erik Cambria, Roy E Welsch, and Francisco Herrera. Distinguishing between facts and opinions for sentiment analysis: Survey and challenges. *Information Fusion*, 44:65 – 77, 2018. 9

[Cen] Australian Cyber Security Centre. https://acsc.gov.au/publications/protect/using-social-media.htm. 91

[cfr] cfr.org. https://www.cfr.org/report/deep-fake-disinformation-steroids. 1, 2

[CG92] George Casella and Edward I George. Explaining the gibbs sampler. *The American Statistician*, 46(3):167–174, 1992. 61

[CGL+18] Juan Cao, Junbo Guo, Xirong Li, Zhiwei Jin, Han Guo, and Jintao Li. Automatic rumor detection on microblogs: A survey. *arXiv preprint arXiv:1807.03505*, 2018. 8, 91, 94

[CHT11] Sarah Cohen, James T Hamilton, and Fred Turner. Computational journalism. *CACM*, 54(10):66–71, 2011. 13

[CHW+08] Michael J Cafarella, Alon Halevy, Daisy Zhe Wang, Eugene Wu, and Yang Zhang. Webtables: exploring the power of tables on the web. In *VLDB*, pages 538–549, 2008. 8

[CLL+18a] Sylvie Cazalens, Philippe Lamarre, Julien Leblay, Ioana Manolescu, and Xavier Tannier. A content management perspective on fact-checking. In *The Web Conference 2018-alternate paper tracks" Journalism, Misinformation and Fact Checking"*, pages 565–574, 2018. 11

[CLL+18b] Sylvie Cazalens, Julien Leblay, Philippe Lamarre, Ioana Manolescu, and Xavier Tannier. Computational fact checking: a content management perspective. In *VLDB*, volume 11, pages 2110–2113, 2018. 11

[CLYY11] Sarah Cohen, Chengkai Li, Jun Yang, and Cong Yu. Computational journalism: A call to arms to database researchers. In *CIDR*, pages 148–151, 2011. 13

[CM09] Olivier Cappé and Eric Moulines. On-line expectation–maximization algorithm for latent data models. *J R Stat Soc Series B Stat Methodol*, 71(3):593–613, 2009. 70

[CMP11] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In *WWW*, pages 675–684, 2011. 12, 42

[CMT18] Tien-Duc Cao, Ioana Manolescu, and Xavier Tannier. Searching for truth in a database of statistics. In *WebDB*, page 4, 2018. 11

[CN14] Feng Chen and Daniel B Neill. Non-parametric scan statistics for event detection and forecasting in heterogeneous social media graphs. In *KDD*, pages 1166–1175, 2014. 12, 13, 32, 36, 37, 38

[CP11] Deepayan Chakrabarti and Kunal Punera. Event summarization using tweets. *ICWSM*, pages 66–73, 2011. 82

[CSG09] Kevin Robert Canini, Lei Shi, and Thomas L Griffiths. Online inference of topics with latent dirichlet allocation. In *AIStats*, pages 65–72, 2009. 21, 81, 85

[CSTC12] Caleb Chen Cao, Jieying She, Yongxin Tong, and Lei Chen. Whom to ask?: Jury selection for decision making tasks on micro-blog services. In *VLDB*, pages 1495–1506, 2012. 16

[CTL+17] Chen Cao, Jiayang Tu, Zheng Liu, Lei Chen, and HV Jagadish. Tuning crowdsourced human computation. In *ICDE*, pages 1021–1032, 2017. 18

[CTY+16] Yi Chang, Jiliang Tang, Dawei Yin, Makoto Yamada, and Yan Liu. Timeline summarization from social media with life cycle models. In *IJCAI*, pages 3698–3704, 2016. 21, 22, 79

[DAK+09] Prabal Dutta, Paul M Aoki, Neil Kumar, Alan Mainwaring, Chris Myers, Wesley Willett, and Allison Woodruff. Common sense: participatory urban sensing using a network of handheld air quality monitors. In *SenSys*, pages 349–350, 2009. 17, 18

[Dat17a] Healthcare Data. http://resources.mpi-inf.mpg.de/impact/peopleondrugs/data.tar.gz, 2017. 71

[dat17b] Snopes dataset. http://tiny.cc/snopesdata, 2017. 56, 71

[DCMT19] Tien Duc Cao, Ioana Manolescu, and Xavier Tannier. Extracting statistical mentions from textual claims to provide trusted content. In *NLDB*, 2019. 9

[Deb01] Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*, volume 16. John Wiley & Sons, 2001. 22

[DGD+17] Sanjib Das, Paul Suganthan GC, AnHai Doan, Jeffrey F Naughton, Ganesh Krishnan, Rohit Deep, Esteban Arcaute, Vijay Raghavendra, and Youngchoon Park. Falcon: Scaling up hands-off crowdsourced entity matching to build cloud services. In *SIGMOD*, pages 1431–1446, 2017. 18

[DGH+14] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *KDD*, pages 601–610, 2014. 8

[DGM19] Ludivine Duroyon, François Goasdoué, and Ioana Manolescu. A Linked Data Model for Facts, Statements and Beliefs. In *MisInfo Workshop*, WWW '19 Companion - Proceedings of the 2019 World Wide Web Conference, San Francisco, United States, 2019. 9

[Die18] Reinhard Diestel. *Graph theory*. Springer Publishing Company, Incorporated, 2018. 35

[DLBS14] Nan Du, Yingyu Liang, Maria Balcan, and Le Song. Influence function learning in information diffusion networks. In *International Conference on Machine Learning*, pages 2016–2024, 2014. 14

[DLV+18] Manh Truong Dang, Anh Vu Luong, Tuyet-Trinh Vu, Quoc Viet Hung Nguyen, Tien Thanh Nguyen, and Bela Stantic. An ensemble system with random projection and dynamic ensemble selection. In *Intelligent Information and Database Systems - 10th Asian Conference, ACIIDS 2018, Dong Hoi City, Vietnam, March 19-21, 2018, Proceedings, Part I*, pages 576–586, 2018. 8

[DNWS17] Chi Thang Duong, Quoc Viet Hung Nguyen, Sen Wang, and Bela Stantic. Provenance-based rumor detection. In *Databases Theory and Applications - 28th Australasian Database Conference, ADC 2017, Brisbane, QLD, Australia, September 25-28, 2017, Proceedings*, pages 125–137, 2017.

[DS14] Norman R Draper and Harry Smith. *Applied regression analysis*, volume 326. John Wiley & Sons, 2014. 14

[DSN09] Kaustav Das, Jeff Schneider, and Daniel B Neill. *Detecting anomalous groups in categorical datasets*. Carnegie Mellon University, School of Computer Science, Machine Learning Department, 2009. 12

[DSRR+16] Christopher De Sa, Alex Ratner, Christopher Ré, Jaeho Shin, Feiran Wang, Sen Wu, and Ce Zhang. Deepdive: Declarative knowledge base construction. *ACM SIGMOD Record*, 45(1):60–67, 2016. 9

[DSS12] Xin Luna Dong, Barna Saha, and Divesh Srivastava. Less is more: Selecting sources wisely for integration. In *VLDB*, pages 37–48, 2012. 10, 54

[DWKDH15] JCF De Winter, Miltos Kyriakidis, Dimitra Dodou, and Riender Happee. Using crowdflower to study the relationship between self-reported violations and traffic accidents. *Procedia Manufacturing*, 3:2518–2525, 2015. 17

[ec19a] Earthsky example claim. http://earthsky.org/human-world/does-eating-turkey-make-you-sleepy, 2019. 56

[ec19b] Kidshealth example claim. http://kidshealth.org/en/kids/turkey-sleepy.html, 2019. 56

[ec19c] Webmd example claim. http://www.webmd.com/food-recipes/the-truth-about-tryptophan, 2019. 56

[eca19] Snopes example claim assessment. http://www.snopes.com/food/ingredient/turkey.asp, 2019. 56

[ECD+04] Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S Weld, and Alexander Yates. Web-scale information extraction in knowitall. In *WWW*, pages 100–110, 2004. 8

[Eig18] Figure Eight. https://www.figure-eight.com, 2018. 17, 77

[Elk08] Charles Elkan. Log-linear models and conditional random fields. *Tutorial notes at CIKM*, 8:1–12, 2008. 58, 59, 60

[ELVZ16] Alessandro Epasto, Silvio Lattanzi, Sergei Vassilvitskii, and Morteza Zadimoghaddam. Submodular optimization over sliding windows. *arXiv preprint arXiv:1610.09984*, 2016. 22

[EMSW14] Patrick Ernst, Cynthia Meng, Amy Siu, and Gerhard Weikum. Knowlife: a knowledge graph for health and life sciences. In *ICDE*, pages 1254–1257, 2014. 21

[Eng] Engadget. https://www.engadget.com/2018/08/21/facebook-rates-user-trustworthiness/. 25

[ESC16] Humaira Ehsan, Mohamed A Sharaf, and Panos K Chrysanthis. Muve: Efficient multi-objective view recommendation for visual data exploration. In *ICDE*, pages 731–742, 2016. 22

[FAEC14] Adrien Friggeri, Lada A Adamic, Dean Eckles, and Justin Cheng. Rumor cascades. In *ICWSM*, pages 101–110, 2014. 9

[Fer73] Thomas S Ferguson. A bayesian analysis of some nonparametric problems. *AOS*, pages 209–230, 1973. 83

[Fis] Fiskkit. https://fiskkit.com/. 11

[FKK+11] Michael J. Franklin, Donald Kossmann, Tim Kraska, Sukriti Ramesh, and Reynold Xin. Crowddb: Answering queries with crowdsourcing. In *SIGMOD*, pages 61–72, 2011. 17

[FKP+12] Michael Fire, Dima Kagan, Rami Puzis, Lior Rokach, and Yuval Elovici. Data mining opportunities in geosocial networks for improving road safety. In *IEEEI*, pages 1–4, 2012. 16

[Fre] Freebase. http://www.freebase.com. 53

[GBKS09] Wolfgang Gatterbauer, Magdalena Balazinska, Nodira Khoussainova, and Dan Suciu. Believe it or not: adding belief annotations to databases. *PVLDB*, 2(1):1–12, 2009. 10, 20

[Giu10] Katherine Del Giudice. Crowdsourcing credibility: The impact of audience feedback on web page credibility. *ASIST*, pages 1–9, 2010. 19

[GKCM14] Aditi Gupta, Ponnurangam Kumaraguru, Carlos Castillo, and Patrick Meier. Tweetcred: Real-time credibility assessment of content on twitter. In *International Conference on Social Informatics*, pages 228–243. Springer, 2014. 12, 44

[GKS01] Sudipto Guha, Nick Koudas, and Kyuseok Shim. Data-streams and histograms. In *STOC*, pages 471–475, 2001. 22

[GKS+13] Avigdor Gal, Michael Katz, Tomer Sagi, Matthias Weidlich, Karl Aberer, Nguyen Quoc Viet Hung, Zoltán Miklós, Eliezer Levy, and Victor Shafran. Completeness and ambiguity of schema cover. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences - Confederated International Conferences: CoopIS, DOA-Trusted Cloud, and ODBASE 2013, Graz, Austria, September 9-13, 2013. Proceedings*, pages 241–258, 2013. 11

[gov] governmenteuropa.eu. https://www.governmenteuropa.eu/cyber-threat-to-digital-democracies/86994/. 1, 91

[GS10] Wolfgang Gatterbauer and Dan Suciu. Data conflict resolution using trust mappings. In *SIGMOD*, pages 219–230, 2010. 10

[GSW04] Sudipto Guha, Kyuseok Shim, and Jungchul Woo. Rehist: Relative error histogram construction algorithms. In *VLDB*, pages 300–311, 2004. 22

[GT04] DMBTL Griffiths and MIJJB Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. *NIPS*, pages 17–24, 2004. 21, 82

[Gua] The Guardian. https://www.theguardian.com/environment/climate-consensus-97-per-cent/2018/jul/25/facebook-video-spreads-climate-denial-misinformation-to-5-million-users. 91

[HA08] Nguyen Quoc Viet Hung and Duong Tuan Anh. An improvement of PAA for dimensionality reduction in large time series databases. In *PRICAI 2008: Trends in Artificial Intelligence, 10th Pacific Rim International Conference on Artificial Intelligence, Hanoi, Vietnam, December 15-19, 2008. Proceedings*, pages 698–707, 2008. 12

[HA13] Nguyen Quoc Viet Hung and Duong Tuan Anh. Using motif information to improve anytime time series classification. In *2013 International Conference on Soft Computing and Pattern Recognition, SoCPaR 2013, Hanoi, Vietnam, December 15-18, 2013*, pages 1–6, 2013. 12

[HBB10] Matthew Hoffman, Francis R Bach, and David M Blei. Online learning for latent dirichlet allocation. In *NIPS*, pages 856–864, 2010. 21, 86, 88

[HBC16] Sara Hajian, Francesco Bonchi, and Carlos Castillo. Algorithmic bias: From discrimination discovery to fairness-aware data mining. In *KDD*, pages 2125–2126, 2016. 2

[HBWP13] Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *JMLR*, pages 1303–1347, 2013. 86, 88

[HD14] Susana Herrera-Damas. Recurring topics in the social media policies of mainstream media. *AJMS*, pages 155–173, 2014. 82

[HDT+17] Nguyen Quoc Viet Hung, Chi Thang Duong, Nguyen Thanh Tam, Matthias Weidlich, Karl Aberer, Hongzhi Yin, and Xiaofang Zhou. Argument discovery via crowdsourcing. *VLDBJ*, 26(4):511–535, 2017. 10, 94

[HJA13] Nguyen Quoc Viet Hung, Hoyoung Jeung, and Karl Aberer. An evaluation of model-based approaches to sensor data compression. *IEEE Trans. Knowl. Data Eng.*, 25(11):2434–2447, 2013. 11

[HLM+13a] Nguyen Quoc Viet Hung, Xuan Hoai Luong, Zoltán Miklós, Thanh Tho Quan, and Karl Aberer. Collaborative schema matching reconciliation. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences - Confederated International Conferences: CoopIS, DOA-Trusted Cloud, and ODBASE 2013, Graz, Austria, September 9-13, 2013. Proceedings*, pages 222–240, 2013. 11

[HLM+13b] Nguyen Quoc Viet Hung, Xuan Hoai Luong, Zoltán Miklós, Thanh Tho Quan, and Karl Aberer. An MAS negotiation support tool for schema matching. In *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '13, Saint Paul, MN, USA, May 6-10, 2013*, pages 1391–1392, 2013. 11

[HN14] Kazi Saidul Hasan and Vincent Ng. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *EMNLP*, pages 751–762, 2014. 59

[HNC+15] Nguyen Quoc Viet Hung, Thanh Tam Nguyen, Vinh Tuan Chau, Tri Kurniawan Wijaya, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. SMART: A tool for analyzing and reconciling schema matching networks. In *31st IEEE International Conference on Data Engineering, ICDE 2015, Seoul, South Korea, April 13-17, 2015*, pages 1488–1491, 2015. 11

[HNLA13a] Nguyen Quoc Viet Hung, Thanh Tam Nguyen, Ngoc Tran Lam, and Karl Aberer. BATC: a benchmark for aggregation techniques in crowdsourcing. In *The 36th International ACM SIGIR conference on research*

and development in Information Retrieval, SIGIR '13, Dublin, Ireland - July 28 - August 01, 2013, pages 1079–1080, 2013. 12

[HNLA13b] Nguyen Quoc Viet Hung, Thanh Tam Nguyen, Ngoc Tran Lam, and Karl Aberer. An evaluation of aggregation techniques in crowdsourcing. In *Web Information Systems Engineering - WISE 2013 - 14th International Conference, Nanjing, China, October 13-15, 2013, Proceedings, Part II*, pages 1–15, 2013. 12

[HNM+14] Nguyen Quoc Viet Hung, Thanh Tam Nguyen, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. Pay-as-you-go reconciliation in schema matching networks. In *IEEE 30th International Conference on Data Engineering, Chicago, ICDE 2014, IL, USA, March 31 - April 4, 2014*, pages 220–231, 2014. 11

[HNMA13] Nguyen Quoc Viet Hung, Thanh Tam Nguyen, Zoltán Miklós, and Karl Aberer. On leveraging crowdsourcing techniques for schema matching networks. In *Database Systems for Advanced Applications, 18th International Conference, DASFAA 2013, Wuhan, China, April 22-25, 2013. Proceedings, Part II*, pages 139–154, 2013. 11

[HNMA14] Nguyen Quoc Viet Hung, Thanh Tam Nguyen, Zoltán Miklós, and Karl Aberer. Reconciling schema matching networks through crowdsourcing. *EAI Endorsed Trans. Collaborative Computing*, 1(2):e2, 2014. 11

[HOA13] Zhicong Huang, Alexandra Olteanu, and Karl Aberer. Credibleweb: a platform for web credibility evaluation. In *CHI*, pages 1887–1892, 2013. 19

[Hof99] Thomas Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, pages 50–57. ACM, 1999. 82

[HSB+16] Bryan Hooi, Hyun Ah Song, Alex Beutel, Neil Shah, Kijung Shin, and Christos Faloutsos. Fraudar: Bounding graph fraud in the face of camouflage. In *KDD*, pages 895–904, 2016. 2

[HSDA14] Nguyen Quoc Viet Hung, Saket Sathe, Chi Thang Duong, and Karl Aberer. Towards enabling probabilistic databases for participatory sensing. In *10th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, CollaborateCom 2014, Miami, Florida, USA, October 22-25, 2014*, pages 114–123, 2014. 11

[HSW+14] Naeemul Hassan, Afroza Sultana, You Wu, Gensheng Zhang, Chengkai Li, Jun Yang, and Cong Yu. Data in, fact out: automated monitoring of facts by factwatcher. *PVLDB*, 7(13):1557–1560, 2014. 13

[HTN+17] Nguyen Quoc Viet Hung, Duong Chi Thang, Thanh Tam Nguyen, Matthias Weidlich, Karl Aberer, Hongzhi Yin, and Xiaofang Zhou. Answer validation for generic crowdsourcing tasks with minimal efforts. *VLDB J.*, 26(6):855–880, 2017.

[HTNA14] Nguyen Quoc Viet Hung, Do Son Thanh, Thanh Tam Nguyen, and Karl Aberer. Privacy-preserving schema reuse. In *Database Systems for*

Advanced Applications - 19th International Conference, DASFAA 2014, Bali, Indonesia, April 21-24, 2014. Proceedings, Part II, pages 234–250, 2014. 11

[HTNA15] Nguyen Quoc Viet Hung, Do Son Thanh, Thanh Tam Nguyen, and Karl Aberer. Tag-based paper retrieval: Minimizing user effort with diversity awareness. In *Database Systems for Advanced Applications - 20th International Conference, DASFAA 2015, Hanoi, Vietnam, April 20-23, 2015, Proceedings, Part I*, pages 510–528, 2015. 10

[HTT+17] Nguyen Quoc Hung, Duong Chi Thang, Nguyen Thanh Tam, Matthias Weidlich, Karl Aberer, Hongzhi Yin, and Xiaofang Zhou. Answer validation for generic crowdsourcing tasks with minimal efforts. *VLDBJ*, 26(6):855–880, 2017. 71

[HTTA13] Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Lam Ngoc Tran, and Karl Aberer. An evaluation of aggregation techniques in crowdsourcing. In *WISE*, pages 1–15, 2013. 19, 22, 77

[HTWA15a] Nguyen Quoc Viet Hung, Duong Chi Thang, Matthias Weidlich, and Karl Aberer. ERICA: expert guidance in validating crowd answers. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, pages 1037–1038, 2015. 11

[HTWA15b] Nguyen Quoc Viet Hung, Duong Chi Thang, Matthias Weidlich, and Karl Aberer. Minimizing efforts in validating crowd answers. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, pages 999–1014, 2015. 11

[HTWA15c] Nguyen Quoc Viet Hung, Duong Chi Thang, Matthias Weidlich, and Karl Aberer. Minimizing efforts in validating crowd answers. In *SIGMOD*, pages 999–1014, 2015. 71, 85

[HVN+18] Nguyen Quoc Viet Hung, Huynh Huu Viet, Thanh Tam Nguyen, Matthias Weidlich, Hongzhi Yin, and Xiaofang Zhou. Computing crowd consensus with partial agreement. *IEEE Trans. Knowl. Data Eng.*, 30(1):1–14, 2018. 8

[HVT+18] Nguyen Quoc Viet Hung, Huynh Huu Viet, Nguyen Thanh Tam, Matthias Weidlich, Hongzhi Yin, and Xiaofang Zhou. Computing crowd consensus with partial agreement. *TKDE*, 30(1):1–14, 2018. 10, 14, 18, 94

[HWC13] Myles Hollander, Douglas A Wolfe, and Eric Chicken. *Nonparametric statistical methods*. John Wiley & Sons, 2013. 82

[HWM+13] Nguyen Quoc Viet Hung, Tri Kurniawan Wijaya, Zoltán Miklós, Karl Aberer, Eliezer Levy, Victor Shafran, Avigdor Gal, and Matthias Weidlich. Minimizing human effort in reconciling match networks. In *Conceptual Modeling - 32th International Conference, ER 2013, Hong-Kong, China, November 11-13, 2013. Proceedings*, pages 212–226, 2013. 11

[HWN+19] Nguyen Quoc Viet Hung, Matthias Weidlich, Thanh Tam Nguyen, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Bela Stantic. Handling probabilistic integrity constraints in pay-as-you-go reconciliation of data models. *Inf. Syst.*, 83:166–180, 2019. 7

[HWT+19] Nguyen Quoc Viet Hung, Matthias Weidlich, Nguyen Thanh Tam, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Bela Stantic. Handling probabilistic integrity constraints in pay-as-you-go reconciliation of data models. *Information Systems*, 83:166–180, 2019. 11

[IHS06] Alexander Ihler, Jon Hutchins, and Padhraic Smyth. Adaptive event detection with time-varying poisson processes. In *KDD*, pages 207–216, 2006. 12

[ILR12] Piotr Indyk, Reut Levi, and Ronitt Rubinfeld. Approximating and testing k-histogram distributions in sub-linear time. In *PODS*, pages 15–22, 2012. 22

[Ioa03] Yannis Ioannidis. The history of histograms (abridged). In *VLDB*, pages 19–30, 2003. 22

[IPC15] Stratos Idreos, Olga Papaemmanouil, and Surajit Chaudhuri. Overview of data exploration techniques. In *SIGMOD*, pages 277–281, 2015. 22

[JFH08] Shawn R. Jeffery, Michael J. Franklin, and Alon Y. Halevy. Pay-as-you-go user feedback for dataspace systems. In *SIGMOD*, pages 847–860, 2008. 19

[JGJS99] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, pages 183–233, 1999. 88

[JKM+98] Hosagrahar Visvesvaraya Jagadish, Nick Koudas, S Muthukrishnan, Viswanath Poosala, Kenneth C Sevcik, and Torsten Suel. Optimal histograms with quality guarantees. In *VLDB*, volume 98, pages 24–27, 1998. 22

[JLLH11] Xin Jin, Cindy Xide Lin, Jiebo Luo, and Jiawei Han. Socialspamguard: A data mining-based spam detection system for social media networks. In *VLDB*, pages 1–458, 2011. 28

[JS94] Björn Jawerth and Wim Sweldens. An overview of wavelet based multiresolution analyses. *SIAM review*, 36(3):377–412, 1994. 22

[JV01] Wen-Hua Ju and Yehuda Vardi. A hybrid high-order markov chain model for computer intrusion detection. *Journal of Computational and Graphical Statistics*, 10(2):277–295, 2001. 12

[KBB80] David Knoke, Peter J Burke, and Peter Burke. *Log-linear models*, volume 20. Sage, 1980. 59

[KCJ17] Sejeong Kwon, Meeyoung Cha, and Kyomin Jung. Rumor detection over varying time windows. *PloS one*, 12(1):e0168344, 2017. 43

[KDM16] Chris Kedzie, Fernando Diaz, and Kathleen McKeown. Real-time web scale event summarization using sequential decision making. In *IJCAI*, pages 3754–3760, 2016. 21

[KFL01] Frank R Kschischang, Brendan J Frey, and H-A Loeliger. Factor graphs and the sum-product algorithm. *TIT*, pages 498–519, 2001. 65

[KJMO10] Dongwoo Kim, Yohan Jo, Il-Chul Moon, and Alice Oh. Analysis of twitter lists as a potential source for discovering latent characteristics of users. In *ACM CHI workshop on microblogging*, page 4, 2010. 45

[KLPM10] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media? In *WWW*, pages 591–600, 2010. 27, 44

[KN+99] Chang-Jin Kim, Charles R Nelson, et al. State-space models with regime switching: classical and gibbs-sampling approaches with applications. *MIT Press Books*, 1, 1999. 61

[KNW+14] Juho Kim, Phu Tran Nguyen, Sarah Weir, Philip J. Guo, Robert C. Miller, and Krzysztof Z. Gajos. Crowdsourcing step-by-step information extraction to enhance existing how-to videos. In *CHI*, pages 4017–4026, 2014. 19

[KNW17] Michal Kakol, Radoslaw Nielek, and Adam Wierzbicki. Understanding and predicting web content credibility using the content credibility corpus. *Information Processing & Management*, 53(5):1043–1061, 2017. 10

[KSG13] Michal Kosinski, David Stillwell, and Thore Graepel. Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the National Academy of Sciences*, 110(15):5802–5805, 2013. 7

[KSKK11] Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E Kraut. Crowdforge: Crowdsourcing complex work. In *UIST*, pages 43–52, 2011. 17

[Kul97] Martin Kulldorff. A spatial scan statistic. *Communications in Statistics-Theory and methods*, pages 1481–1496, 1997. 36

[LDL+13] Xian Li, Xin Luna Dong, Kenneth Lyons, Weiyi Meng, and Divesh Srivastava. Truth finding on the deep web: is the problem solved? In *VLDB*, pages 97–108, 2013. 11

[LDLL17] Furong Li, Xin Luna Dong, Anno Langen, and Yang Li. Knowledge verification for long-tail verticals. In *VLDB*, pages 1370–1381, 2017. 9, 10

[Lei50] Harvey Leibenstein. Bandwagon, snob, and veblen effects in the theory of consumers' demand. *The quarterly journal of economics*, 64(2):183–207, 1950. 13

[LGMN12] Jens Lehmann, Daniel Gerber, Mohamed Morsey, and Axel-Cyrille Ngonga Ngomo. Defacto-deep fact validation. In *ISWC*, pages 312–327, 2012. 54

[LKC99] Ju-Hong Lee, Deok-Hwan Kim, and Chin-Wan Chung. Multi-dimensional selectivity estimation using compressed histogram information. In *SIGMOD*, volume 28, pages 205–214, 1999. 22

[LMT18] Julien Leblay, Ioana Manolescu, and Xavier Tannier. Computational fact-checking: Problems, state of the art, and perspectives. In *The Web Conference*, 2018. 11

[LMY11] Xian Li, Weiyi Meng, and Clement Yu. T-verifier: Verifying truthfulness of fact statements. In *ICDE*, pages 63–74, 2011. 9

[LNL+15] Xiaomo Liu, Armineh Nourbakhsh, Quanzhi Li, Rui Fang, and Sameena Shah. Real-time rumor debunking on twitter. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1867–1870. ACM, 2015. 2, 12

[LS08] Florian Laws and Hinrich Schätze. Stopping criteria for active learning of named entity recognition. In *ICCL*, pages 465–472, 2008. 20

[LSD+17] Yaguang Li, Han Su, Ugur Demiryurek, Bolong Zheng, Tieke He, and Cyrus Shahabi. Pare: A system for personalized route guidance. In *WWW*, pages 637–646, 2017. 19

[LWK08] Chih-Jen Lin, Ruby C Weng, and S Sathiya Keerthi. Trust region newton method for logistic regression. *JMLR*, pages 627–650, 2008. 61, 70

[LWZF16] Guoliang Li, Jiannan Wang, Yudian Zheng, and Michael J Franklin. Crowdsourced data management: A survey. *TKDE*, 28(9):2296–2319, 2016. 19

[LZZ+18] Yu Liu, Hantian Zhang, Luyuan Zeng, Wentao Wu, and Ce Zhang. Mlbench: benchmarking machine learning services against human experts. In *VLDB*, pages 1220–1232, 2018. 18

[Man17] Ioana Manolescu. Contentcheck: Content management techniques and tools for fact-checking. *ERCIM News*, 2017. 11

[Man19] Ioana Manolescu. Computational fact-checking: Problems, state of the art and perspectives. In *19e Conférence Francophone sur l'Extraction et Gestion de Connaissances (EGC)*, 2019. 11

[MBK+15] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrák, and Andreas Krause. Lazier than lazy greedy. In *AAAI*, pages 1813–1818, 2015. 22

[Met13] Carl Metzgar. Confirmation bias: A ubiquitous phenomenon in many guises. *Professional Safety*, 58(9):44, 2013. 13

[MGAM16] Panagiotis Mavridis, David Gross-Amblard, and Zoltán Miklós. Using hierarchical skills for optimized task assignment in knowledge-intensive crowdsourcing. In *WWW*, pages 843–853, 2016. 19

[MGM+16] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. Detecting rumors from microblogs with recurrent neural networks. In *IJCAI*, pages 3818–3824, 2016. 12, 24

[MGW+15] Jing Ma, Wei Gao, Zhongyu Wei, Yueming Lu, and Kam-Fai Wong. Detect rumors using time series of social context information on microblogging websites. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1751–1754. ACM, 2015. 12

[MGW17] Jing Ma, Wei Gao, and Kam-Fai Wong. Detect rumors in microblog posts using propagation structure via kernel learning. In *ACL*, pages 708–717, 2017. 12

[MK10] Michael Mathioudakis and Nick Koudas. Twittermonitor: trend detection over the twitter stream. In *SIGMOD*, pages 1155–1158, 2010. 79

[MKSK13] Baharan Mirzasoleiman, Amin Karbasi, Rik Sarkar, and Andreas Krause. Distributed submodular maximization: Identifying representative elements in massive data. In *Advances in Neural Information Processing Systems*, pages 2049–2057, 2013. 22

[MPLC13] Fred Morstatter, Jurgen Pfeffer, Huan Liu, and Kathleen M Carley. Is the sample good enough? comparing data from twitter's streaming api with twitter's firehose. In *ICWSM*, pages 400–408, 2013. 31, 35, 44, 83

[MRS+09] Min Mun, Sasank Reddy, Katie Shilton, Nathan Yau, Jeff Burke, Deborah Estrin, Mark Hansen, Eric Howard, Ruth West, and Péter Boda. Peir, the personal environmental impact report, as a platform for participatory sensing systems research. In *MobiSys*, pages 55–68, 2009. 19

[MS13] Krikamol Muandet and Bernhard Schölkopf. One-class support measure machines for group anomaly detection. *arXiv preprint arXiv:1303.0309*, 2013. 12

[MSF+14] Barzan Mozafari, Purna Sarkar, Michael Franklin, Michael Jordan, and Samuel Madden. Scaling up crowd-sourcing to very large datasets: A case for active learning. In *VLDB*, pages 125–136, 2014. 20, 71

[MW15] Subhabrata Mukherjee and Gerhard Weikum. Leveraging joint interactions for credibility analysis in news communities. In *CIKM*, pages 353–362, 2015. 10, 61

[MWDNM14a] Subhabrata Mukherjee, Gerhard Weikum, and Cristian Danescu-Niculescu-Mizil. People on drugs: Credibility of user statements in health communities. In *KDD*, pages 65–74, 2014. 9

[MWDNM14b] Subhabrata Mukherjee, Gerhard Weikum, and Cristian Danescu-Niculescu-Mizil. People on drugs: credibility of user statements in health communities. In *KDD*, pages 65–74, 2014. 21, 44, 54, 71, 80, 84

[MZ11] Hamid Mousavi and Carlo Zaniolo. Fast and accurate computation of equi-depth histograms over data streams. In *EDBT*, pages 69–80, 2011. 22

[NBW18] Aleksandra Nabożny, Bartłomiej Balcerzak, and Adam Wierzbicki. Automatic credibility assessment of popular medical articles available online. In *SocInfo*, pages 215–223, 2018. 10

[NDG12] Anis Najar, Ludovic Denoyer, and Patrick Gallinari. Predicting information diffusion on social networks with partial knowledge. In *Proceedings of the 21st International Conference on World Wide Web*, pages 1197–1204, 2012. 14

[NDN+17] Quoc Viet Hung Nguyen, Chi Thang Duong, Thanh Tam Nguyen, Matthias Weidlich, Karl Aberer, Hongzhi Yin, and Xiaofang Zhou. Argument discovery via crowdsourcing. *JVLDB*, pages 1–25, 2017. 21

[NDW+17] Thanh Tam Nguyen, Chi Thang Duong, Matthias Weidlich, Hongzhi Yin, and Quoc Viet Hung Nguyen. Retaining data from streams of social platforms with minimal regret. In *IJCAI*, pages 2850–2856, 2017. 69

[Nei12] Daniel B Neill. Fast subset scan for spatial pattern detection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):337–360, 2012. 38

[Net] NetworkWorld. https://www.networkworld.com/article/2235277/data-center/data-center-fact-checking-the-fact-checkers-snopes-com-gets-an-a.html. 43

[New] BuzzFeed News. http://tiny.cc/31my2y. 1

[NHQ12] Thanh Tam Nguyen, Nguyen Quoc Viet Hung, and Thanh Tho Quan. A framework to combine multiple matchers for pair-wise schema matching. In *2012 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), Ho Chi Minh City, Vietnam, February 27 - March 1, 2012*, pages 1–6, 2012. 12

[NHWA15] Thanh Tam Nguyen, Nguyen Quoc Viet Hung, Matthias Weidlich, and Karl Aberer. Result selection and summarization for web table search. In *31st IEEE International Conference on Data Engineering, ICDE 2015, Seoul, South Korea, April 13-17, 2015*, pages 231–242, 2015. 11

[NMD12] Jeffrey Nichols, Jalal Mahmud, and Clemens Drews. Summarizing sporting events using twitter. In *IUI*, pages 189–198, 2012. 22

[NNL+19] Tien Thanh Nguyen, Thi Thu Thuy Nguyen, Anh Vu Luong, Quoc Viet Hung Nguyen, Alan Wee-Chung Liew, and Bela Stantic. Multi-label classification via label correlation and first order feature dependance in a data stream. *Pattern Recognition*, 90:35–51, 2019. 7

[NNM+14] Quoc Viet Hung Nguyen, Thanh Tam Nguyen, Zoltán Miklós, Karl Aberer, Avigdor Gal, and Matthias Weidlich. Pay-as-you-go reconciliation in schema matching networks. In *ICDE*, pages 220–231, 2014. 11, 71

[NNWA15] Thanh Tam Nguyen, Quoc Viet Hung Nguyen, Matthias Weidlich, and Karl Aberer. Result selection and summarization for web table search. In *ICDE*, pages 231–242, 2015. 11, 22

[NPN+19] Thanh Tam Nguyen, Thanh Cong Phan, Quoc Viet Hung Nguyen, Karl Aberer, and Bela Stantic. Maximal fusion of facts on the web with credibility guarantee. *Information Fusion*, 48:55–66, 2019. 7

[NVN+18] Quoc Viet Hung Nguyen, Huynh Huu Viet, Thanh Tam Nguyen, Matthias Weidlich, Hongzhi Yin, and Xiaofang Zhou. Computing crowd consensus with partial agreement. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018*, pages 1749–1750, 2018. 9

[NW81] George L Nemhauser and Laurence A Wolsey. Maximizing submodular set functions: formulations and analysis of algorithms. *North-Holland Mathematics Studies*, pages 279–301, 1981. 68, 69, 85

[NWZ+19] Thanh Tam Nguyen, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Quoc Viet Hung Nguyen, and Bela Stantic. From anomaly detection to rumour detection using data streams of social platforms. *PVLDB*, 12(9):1016–1029, 2019. 7

[NYW+19] Thanh Tam Nguyen, Hongzhi Yin, Matthias Weidlich, Bolong Zheng, Quoc Viet Hung Nguyen, and Bela Stantic. User guidance for efficient fact checking. *PVLDB*, 12(8):850–863, 2019. 7

[NZW+18a] Quoc Viet Hung Nguyen, Kai Zheng, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Thanh Tam Nguyen, and Bela Stantic. What-if analysis with conflicting goals: Recommending data ranges for exploration. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018*, pages 89–100, 2018. 9

[NZW+18b] Quoc Viet Hung Nguyen, Kai Zheng, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Thanh Tam Nguyen, and Bela Stantic. What-if analysis with conflicting goals: Recommending data ranges for exploration. In *ICDE*, pages 89–100, 2018. 94

[oC] University of Canberra. http://tiny.cc/o3my2y. 1

[OCDA15] Alexandra Olteanu, Carlos Castillo, Nicholas Diakopoulos, and Karl Aberer. Comparing events coverage in online news and social media: The case of climate change. In *ICWSM*, pages 288–297, 2015. 25

[OCDV14] Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *ICWSM*, pages 376–385, 2014. 25

[Oko09] Ory Okolloh. Ushahidi, or'testimony': Web 2.0 tools for crowdsourcing crisis information. *Participatory learning and action*, pages 65–70, 2009. 16

[Olt16] Alexandra Olteanu. Probing the limits of social data. Technical report, EPFL, 2016. 7

[OPLA13] Alexandra Olteanu, Stanislav Peshterliev, Xin Liu, and Karl Aberer. Web credibility: Features exploration and credibility prediction. In *ECIR*, pages 557–568, 2013. 71

[PBKL14] Vu Pham, Théodore Bluche, Christopher Kermorvant, and Jérôme Louradour. Dropout improves recurrent neural networks for handwriting recognition. In *ICFHR*, pages 285–290, 2014. 29

[PCBH17] Soujanya Poria, Erik Cambria, Rajiv Bajpai, and Amir Hussain. A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, 37:98–125, 2017. 9

[PDA17] Barbara Pes, Nicoletta Dessì, and Marta Angioni. Exploiting the ensemble paradigm for stable feature selection: A case study on high-dimensional genomic data. *Information Fusion*, 35:132–147, 2017. 20

[PGQdC17] Pablo Pérez-Gállego, José Ramón Quevedo, and Juan José del Coz. Using ensembles for problems with characterizable changes in data distribution: A case study on quantification. *Information Fusion*, 34:87–100, 2017. 20

[PH16] Douglas Alves Peixoto and Nguyen Quoc Viet Hung. Scalable and fast top-k most similar trajectories search using mapreduce in-memory. In *Databases Theory and Applications - 27th Australasian Database Conference, ADC 2016, Sydney, NSW, Australia, September 28-29, 2016, Proceedings*, pages 228–241, 2016. 10

[PHIS96] Viswanath Poosala, Peter J Haas, Yannis E Ioannidis, and Eugene J Shekita. Improved histograms for selectivity estimation of range predicates. In *SIGMOD*, volume 25, pages 294–305, 1996. 22

[PI97] Viswanath Poosala and Yannis Ioannidis. Selectivity estimation without the attribute value independence assumption. In *VLDB*, volume 97, pages 486–495, 1997. 22

[PMSW17] Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. Where the truth lies: Explaining the credibility of emerging claims on the web and social media. In *WWW Companion*, pages 1003–1012, 2017. 11, 60, 71

[PR13] Jeff Pasternack and Dan Roth. Latent credibility analysis. In *WWW*, pages 1009–1020, 2013. 10

[PROA12] Thanasis G. Papaioannou, Jean-Eudes Ranvier, Alexandra Olteanu, and Karl Aberer. A decentralized recommender system for effective web credibility assessment. In *CIKM*, pages 704–713, 2012. 9

[PSH+18] Douglas Alves Peixoto, Han Su, Nguyen Quoc Viet Hung, Bela Stantic, Bolong Zheng, and Xiaofang Zhou. Concept for evaluation of techniques for trajectory distance measures. In *19th IEEE International Conference on Mobile Data Management, MDM, 2018, Aalborg, Denmark, June 25-28, 2018*, pages 276–277, 2018. 9

[PTHS18] Thanh Cong Phan, Nguyen Thanh Toan, Nguyen Quoc Viet Hung, and Bela Stantic. Minimizing efforts in reconciling participatory sensing data. In *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, WIMS 2018, Novi Sad, Serbia, June 25-27, 2018*, pages 49:1–49:11, 2018. 10

[PZH+18] Douglas Alves Peixoto, Xiaofang Zhou, Nguyen Quoc Viet Hung, Dan He, and Bela Stantic. A system for spatial-temporal trajectory data integration and representation. In *Database Systems for Advanced Applications - 23rd International Conference, DASFAA 2018, Gold Coast, QLD, Australia, May 21-24, 2018, Proceedings, Part II*, pages 807–812, 2018. 9

[QLNY18] Tieyun Qian, Bei Liu, Nguyen Quoc Viet Hung, and Hongzhi Yin. Spatiotemporal representation learning for translation-based poi recommendation. *TOIS*, 2018. 10

[Rea90] James Reason. *Human error*. Cambridge university press, 1990. 65

[Rey13] Matthew G Reyes. Covariance and entropy in markov random fields. In *ITA*, pages 1–6, 2013. 62

[RIS+10] J. Ross, L. Irani, M. Silberman, A. Zaldivar, and B. Tomlinson. Who are the crowdworkers?: shifting demographics in mechanical turk. In *CHI*, pages 2863–2872, 2010. 14

[RN03] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2003. 63, 71

[RN09] Matthew G Reyes and David L Neuhoff. Entropy bounds for a markov random subfield. In *ISIT*, pages 309–313, 2009. 62

[Rub10] Victoria L Rubin. On deception and deception detection: Content analysis of computer-mediated stated beliefs. In *ASIST*, page 32, 2010. 13

[RZR07] Iyad Rahwan, Fouad Zablith, and Chris Reed. Laying the foundations for a world wide argument web. *Artificial intelligence*, 171(10-15):897–921, 2007. 11

[SAH12] Yizhou Sun, Charu C Aggarwal, and Jiawei Han. Relation strength-aware clustering of heterogeneous information networks with incomplete attributes. In *VLDB*, pages 394–405, 2012. 28

[SCA19] Panayiotis Smeros, Carlos Castillo, and Karl Aberer. Scilens: Evaluating the quality of scientific news articles using social media and scientific literature indicators. In *WWW*, pages 1747–1758, 2019. 10

[Sch09] Mikkel Schmidt. Linearly constrained bayesian matrix factorization for blind source separation. In *NIPS*, pages 1624–1632, 2009. 61

[SDJ+01] Matthias Schonlau, William DuMouchel, Wen-Hua Ju, Alan F Karr, Martin Theus, and Yehuda Vardi. Computer intrusion: Detecting masquerades. *Statistical science*, pages 58–74, 2001. 12

[Set09] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009. 20

[Set12] Burr Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–114, 2012. 20

[SH12] Yizhou Sun and Jiawei Han. Mining heterogeneous information networks: principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 3(2):1–159, 2012. 9

[SLZ+17] Chuan Shi, Yitong Li, Jiawei Zhang, Yizhou Sun, and S Yu Philip. A survey of heterogeneous information network analysis. *TKDE*, 29(1):17–37, 2017. 9, 26

[Snoa] Snopes. https://www.snopes.com/fact-check/las-vegas-shooting-rumors-hoaxes-and-conspiracy-theories/. 32

[Snob] Snopes. https://www.snopes.com/fact-check/trump-aid-puerto-rico/. 43

[Snoc] Snopes. http://tiny.cc/las-vegas-shooting. 50

[Snod] Snopes. http://tiny.cc/p1s2qy. 44

[Spi] Spinn3r. http://docs.spinn3r.com/. 43

[SRYD14] Chong Sun, Narasimhan Rampalli, Frank Yang, and AnHai Doan. Chimera: Large-scale classification using machine learning, rules, and crowdsourcing. In *VLDB*, 2014. 18

[SS17] Edouard Ngor Sarr and Ousmane Sall. Automation of fact-checking: State of the art, obstacles and perspectives. In *DASC*, pages 1314–1317, 2017. 9, 13

[SSW+17] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36, 2017. 12

[STVB16] Mehdi Samadi, Partha Pratim Talukdar, Manuela M Veloso, and Manuel Blum. Claimeval: Integrated and flexible framework for claim evaluation using credibility of sources. In *AAAI*, pages 222–228, 2016. 9, 20

[SZN13] Skyler Speakman, Yating Zhang, and Daniel B Neill. Dynamic pattern detection with temporal consistency and connectivity constraints. In *ICDM*, pages 697–706, 2013. 41

[TF13] Io Taxidou and Peter Fischer. Realtime analysis of information diffusion in social media. In *VLDB*, pages 1416–1421, 2013. 28

[THL+15] Immanuel Trummer, Alon Halevy, Hongrae Lee, Sunita Sarawagi, and Rahul Gupta. Mining subjective properties on the web. In *SIGMOD*, pages 1745–1760, 2015. 10

[TJBB06] Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. Hierarchical dirichlet processes. *JASA*, pages 1566–1581, 2006. 83

[TK02] Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *JMLR*, pages 45–66, 2002. 20

[TM07] Nava Tintarev and Judith Masthoff. A survey of explanations in recommender systems. In *ICDEW*, pages 801–810, 2007. 94

[TNHA15] Duong Chi Thang, Thanh Tam Nguyen, Nguyen Quoc Viet Hung, and Karl Aberer. An evaluation of diversification techniques. In *Database and Expert Systems Applications - 26th International Conference, DEXA 2015, Valencia, Spain, September 1-4, 2015, Proceedings, Part II*, pages 215–231, 2015. 11

[TPN+18] Nguyen Thanh Toan, Thanh Cong Phan, Thanh Tam Nguyen, Nguyen Quoc Viet Hung, and Bela Stantic. Diversifying group recommendation. *IEEE Access*, 6:17776–17786, 2018. 8

[TPT+18] Nguyen Thanh Toan, Thanh Cong Phan, Duong Chi Thang, Nguyen Quoc Viet Hung, and Bela Stantic. Bootstrapping uncertainty in schema covering. In *Databases Theory and Applications - 29th Australasian Database Conference, ADC 2018, Gold Coast, QLD, Australia, May 24-27, 2018, Proceedings*, pages 336–342, 2018. 8

[TQW+15] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *WWW*, pages 1067–1077, 2015. 9

[TTFS18] Luan Tran, Hien To, Liyue Fan, and Cyrus Shahabi. A real-time framework for task assignment in hyperlocal spatial crowdsourcing. *TIST*, 9(3):37, 2018. 14, 18

[Tuf14] Zeynep Tufekci. Big questions for social media big data: Representativeness, validity and other methodological pitfalls. In *Eighth International AAAI Conference on Weblogs and Social Media*, 2014. 7

[Tum] Tumblr. http://www.tumblr.com. 7

[TVF+17] Behzad Tabibian, Isabel Valera, Mehrdad Farajtabar, Le Song, Bernhard Schölkopf, and Manuel Gomez-Rodriguez. Distilling information reliability and source trustworthiness from digital traces. In *WWW*, pages 847–855, 2017. 10

[Twi] Twitter. http://www.twitter.com. 7

[TWT+17] Nguyen Thanh Tam, Matthias Weidlich, Duong Chi Thang, Hongzhi Yin, and Nguyen Quoc Viet Hung. Retaining data from streams of social platforms with minimal regret. In *IJCAI*, pages 2850–2856, 2017. 10

[TWY+19] Nguyen Thanh Tam, Matthias Weidlich, Hongzhi Yin, Bolong Zheng, Nguyen Quoc Viet Hung, and Bela Stantic. User guidance for efficient fact checking. In *VLDB*, 2019. 14

[TWZ+19] Nguyen Thanh Tam, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Nguyen Quoc Viet Hung, and Bela Stantic. From anomaly detection to rumour detection using data streams of social platforms. In *VLDB*, 2019. 12

[VAD04] Luis Von Ahn and Laura Dabbish. Labeling images with a computer game. In *SIGCHI*, pages 319–326, 2004. 15

[VAMM+08] Luis Von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. recaptcha: Human-based character recognition via web security measures. *Science*, pages 1465–1468, 2008. 15

[Ver] The Verge. https://www.theverge.com/2018/8/21/17763886/ facebook-trust-ratings-fake-news-reporting-score. 25

[VG14] Sudheendra Vijayanarasimhan and Kristen Grauman. Large-scale live active learning: Training object detectors with crawled data and crowds. *CVPR*, pages 97–114, 2014. 18

[VLCH18] Ana Valdivia, M Victoria Luzón, Erik Cambria, and Francisco Herrera. Consensus vote models for detecting and filtering neutrality in sentiment analysis. *Information Fusion*, 44:126–135, 2018. 9

[VR15] Soroush Vosoughi and Deb Roy. A human-machine collaborative system for identifying rumors on twitter. In *ICDMW*, pages 47–50, 2015. 21

[VRA18] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018. 1, 8, 21, 24, 25, 45, 46, 71, 91, 94

[WAL+14] You Wu, Pankaj K Agarwal, Chengkai Li, Jun Yang, and Cong Yu. Toward computational fact-checking. In *VLDB*, pages 589–600, 2014. 54

[WHMO13] Miao Wang, Viliam Holub, John Murphy, and Patrick O'Sullivan. High volumes of event stream indexing and efficient multi-keyword searching for cloud monitoring. *Future Generation Computer Systems*, 29(8):1943–1962, 2013. 21

[Wie18] Adam Wierzbicki. *Web Content Credibility*. Springer, 2018. 10

[Wik] Wikipedia. http://wikipedia.org. 7

[Wik17] Wikidata. http://tiny.cc/wikidata, 2017. 71

[WKLA12] Dong Wang, Lance Kaplan, Hieu Le, and Tarek Abdelzaher. On truth discovery in social sensing: A maximum likelihood estimation approach. In *IPSN*, pages 233–244, 2012. 10

[WP15] Yichen Wang and Aditya Pal. Detecting emotions in social media: A constrained optimization approach. In *IJCAI*, pages 996–1002, 2015. 21

[WSA15] Yi Wang, Yu Su, and Gagan Agrawal. A novel approach for approximate aggregations over arrays. In *SSDBM*, pages 4:1–4:12, 2015. 22

[WT15] Shihan Wang and Takao Terano. Detecting rumor patterns in streaming social media. In *Big Data*, pages 2709–2715, 2015. 12

[WYZ15] Ke Wu, Song Yang, and Kenny Q Zhu. False rumors detection on sina weibo by propagation structures. In *ICDE*, pages 651–662, 2015. 12

[XPS+11] Liang Xiong, Barnabás Póczos, Jeff G Schneider, Andrew Connolly, and Jake VanderPlas. Hierarchical probabilistic models for group anomaly detection. In *International Conference on Artificial Intelligence and Statistics*, pages 789–797, 2011. 12

[YAG] YAGO. http://www.mpi-inf.mpg.de/yago. 53

[YCB+07] Alexander Yates, Michael Cafarella, Michele Banko, Oren Etzioni, Matthew Broadhead, and Stephen Soderland. Textrunner: open information extraction on the web. In *NAACL-HLT*, pages 25–26, 2007. 9

[YCS+17] Hongzhi Yin, Hongxu Chen, Xiaoshuai Sun, Hao Wang, Yang Wang, and Quoc Viet Hung Nguyen. SPTF: A scalable probabilistic tensor factorization model for semantic-aware behavior prediction. In *2017 IEEE International Conference on Data Mining, ICDM 2017, New Orleans, LA, USA, November 18-21, 2017*, pages 585–594, 2017.

[YEN+11] Mohamed Yakout, Ahmed K Elmagarmid, Jennifer Neville, Mourad Ouzzani, and Ihab F Ilyas. Guided data repair. In *VLDB*, pages 279–289, 2011. 11, 19

[YHL15] Rose Yu, Xinran He, and Yan Liu. Glad: group anomaly detection in social media analysis. *TKDD*, 10(2):18, 2015. 12

[YHZ+16] Hongzhi Yin, Zhiting Hu, Xiaofang Zhou, Hao Wang, Kai Zheng, Nguyen Quoc Viet Hung, and Shazia Wasim Sadiq. Discovering interpretable geo-social communities for user behavior prediction. In *32nd IEEE International Conference on Data Engineering, ICDE 2016, Helsinki, Finland, May 16-20, 2016*, pages 942–953, 2016. 10

[YJP17] Zheng Yan, Xuyang Jing, and Witold Pedrycz. Fusing and mining opinions for reputation generation. *Information Fusion*, 36:172–184, 2017. 9

[YKA16] Junting Ye, Santhosh Kumar, and Leman Akoglu. Temporal opinion spam detection by multivariate indicative signals. In *ICWSM*, pages 743–746, 2016. 25

[YLC16] Dingqi Yang, Bin Li, and Philippe Cudré-Mauroux. Poisketch: Semantic place labeling over user activity streams. In *IJCAI*, pages 2697–2703, 2016. 21

[YLLL17] Bo Yang, Yu Lei, Jiming Liu, and Wenjie Li. Social collaborative filtering by trust. *TPAMI*, 39(8):1633–1647, 2017. 10

[YLYY12] Fan Yang, Yang Liu, Xiaohui Yu, and Min Yang. Automatic detection of rumor on sina weibo. In *KDD*, page 13, 2012. 12, 42

[YQW+16] Rose Yu, Huida Qiu, Zhen Wen, ChingYung Lin, and Yan Liu. A survey on social media anomaly detection. *ACM SIGKDD Explorations Newsletter*, 18(1):1–14, 2016. 12

[YZC+16] Hongzhi Yin, Xiaofang Zhou, Bin Cui, Hao Wang, Kai Zheng, and Nguyen Quoc Viet Hung. Adapting to user interest drift for POI recommendation. *IEEE Trans. Knowl. Data Eng.*, 28(10):2566–2581, 2016. 10

[YZN+18] Hongzhi Yin, Lei Zou, Quoc Viet Hung Nguyen, Zi Huang, and Xiaofang Zhou. Joint event-partner recommendation in event-based social networks. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018*, pages 929–940, 2018. 9

[ZAB+17] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. Detection and resolution of rumours in social media: A survey. *arXiv preprint arXiv:1704.00656*, 2017. 12

[ZAB+18] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. Detection and resolution of rumours in social media: A survey. *CSUR*, 51(2):32, 2018. 1, 9, 12, 25, 42, 91

[ZBG13] Ke Zhai and Jordan L Boyd-Graber. Online latent dirichlet allocation with infinite vocabulary. *ICML*, pages 561–569, 2013. 21, 82, 83, 84

[ZE01] Bianca Zadrozny and Charles Elkan. Learning and making decisions when costs and probabilities are both unknown. In *KDD*, pages 204–213, 2001. 20

[Zea18] Xinyi Zhou et. al. Fake news: A survey of research, detection methods, and opportunities. *arXiv:1812.00315*, 2018. 1, 2, 8, 12, 13, 14, 91, 93

[ZL18] Gensheng Zhang and Chengkai Li. Maverick: a system for discovering exceptional facts from knowledge graphs. In *VLDB*, pages 1934–1937, 2018. 71

[ZLP17] Arkaitz Zubiaga, Maria Liakata, and Rob Procter. Exploiting context for rumour detection in social media. In *Socinfo*, pages 109–123, 2017. 21

[ZLR05] Xiaojin Zhu, John Lafferty, and Ronald Rosenfeld. *Semi-supervised learning with graphs*. Carnegie Mellon University, 2005. 20

[ZRGH12] Bo Zhao, Benjamin IP Rubinstein, Jim Gemmell, and Jiawei Han. A bayesian approach to discovering truth from conflicting sources for data integration. In *VLDB*, pages 550–561, 2012. 10

[ZRH+16] Hao Zhuang, Rameez Rahman, Xia Hu, Tian Guo, Pan Hui, and Karl Aberer. Data summarization with social contexts. In *CIKM*, pages 397–406, 2016. 22, 79, 80, 82, 83, 84, 85, 88

[ZRM15] Zhe Zhao, Paul Resnick, and Qiaozhu Mei. Enquiring minds: Early detection of rumors in social media from enquiry posts. In *WWW*, pages 1395–1405, 2015. 9, 12, 24, 42

[ZZJ+19] Bolong Zheng, Kai Zheng, Christian S. Jensen, Nguyen Quoc Viet Hung, Han Su, Guohui Li, and Xiaofang Zhou. Answering why-not group spatial keyword queries (extended abstract). In *35th IEEE International Conference on Data Engineering, ICDE 2019, Macao, China, April 8-11, 2019*, pages 2155–2156, 2019. 8

[ZZS+19] Bolong Zheng, Kai Zheng, Peter Scheuermann, Xiaofang Zhou, Quoc Viet Hung Nguyen, and Chenliang Li. Searching activity trajectory with keywords. *World Wide Web*, 22(3):967–1000, 2019. 8

[ZZZ+17] Fan Zhang, Wenjie Zhang, Ying Zhang, Lu Qin, and Xuemin Lin. Olak: an efficient algorithm to prevent unraveling in social networks. In *VLDB*, pages 649–660, 2017. 28

Appendix A

Curriculum Vitae

Nguyen Thanh Tam

PhD., EPFL, Switzerland

School of Computer and Communication Sciences
EPFL, Switzerland
☎ *+41 (78) 929 9322*
✉ *thanhtamlhp@gmail.com*

Research Interests

My research interests include Data Filtering, Trust Management, Crowdsourcing, Machine Learning, and Blockchain, with special focus on data lake, data networks, and data streams.

$\sim$ 20 publications

10 CORE/ERA Tier A^* & 9 Tier A publications

h-index: 12, **citation:** 529

Education

09.2014–now **Ph.D.**, *Ecole Polytechnique de Lausanne (EPFL)*, Switzerland.

09.2012–02.2014 **Master of Computer Science**, *EPFL*, Switzerland.

09.2007–02.2012 **Bachelor of Computer Science and Engineering**, *Hochiminh City University of Technology (HCMUT)*, Vietnam.

Work Experience

2013 **Software Intern**, *Akselos S.A., Parc Scientifique, EPFL*, Switzerland.

2011–2012 **Research Assistant**, *Distributed Information Systems Laboratory, EPFL*, Switzerland.

2010 **Student Intern**, *Distributed Information Systems Laboratory, EPFL*, Switzerland.

Honors

2013 Best student paper award - DASFAA 2013

2012 Silver medal - ACM programming contest SWERC 2012
1st rank - EPFL programing contest

2011–2012 "Research Scholar" for Master study at EPFL

Research Projects

09.2010-02.2013 **EU-FP7 NisB** – developing cost-driven models for integrating and reusing data
Researcher

09.2010-08.2014 **EU-FP7 PlanetData** – establishing an interdisciplinary repository by linking, managing, and benchmarking Web data
Researcher

01.2014-01.2018 **Swiss-NSF ScienceWise** – building taxonomy for physics, life sciences, digital humanities and information technologies
Researcher

Service

External Reviewer — JVLDB, TKDE, EDBT, DASFAA, WWW, PAKDD, ICDE, SIGIR, ISWC, WISE, CIKM, VLDB, BPM

Teaching Assistant — Distributed Information Systems (2017,2018), Programming 1, Programming 2, Discrete Structures

Selected Publications

Journal Articles

TIST 2019 — Nguyen Thanh Tam, Hongzhi Yin, Bolong Zheng, Nguyen Quoc Viet Hung, Bela Stantic. **Efficient User Guidance for Validating Participatory Sensing Data**. In TIST 2019. (IF 10.47)

IS 2019 — Nguyen Quoc Viet Hung, Matthias Weidlich, Nguyen Thanh Tam, Zoltan Miklos, Karl Aberer, Avigdor Gal, Bela Stantic. **Handling Probabilistic Integrity Constraints in Pay-as-you-go Reconciliation of Data Models**. In Information Systems Journal 2019. (IF 4.267)

INFFUS 2018 (A) — Thanh Tam Nguyen, Thanh Cong Phan, Quoc Viet Hung Nguyen, Karl Aberer, Bela Stantic. **Maximal Fusion of Facts on the Web with Credibility Guarantee**. In INFFUS 2018. (IF 6.64)

IEEE Access 2018 — Nguyen Thanh Toan, Phan Thanh Cong, Nguyen Thanh Tam, Nguyen Quoc Viet Hung, Bela Stantic. **Diversifying Group Recommendation**. In IEEE Access 2018. (IF 3.244)

JVLDB 2017 (A^*) — Nguyen Quoc Viet Hung, Duong Chi Thang, Nguyen Thanh Tam, Matthias Weidlich, Karl Aberer, Hongzhi Yin, Xiaofang Zhou. **Answer Validation for Generic Crowdsourcing Tasks with Minimal Efforts**. In JVLDB 2017. (IF 4.269)

JVLDB 2017 (A^*) — Nguyen Quoc Viet Hung, Duong Chi Thang, Nguyen Thanh Tam, Matthias Weidlich, Karl Aberer, Hongzhi Yin, Xiaofang Zhou. **Argument Discovery via Crowdsourcing**. In JVLDB 2017. (IF 4.269)

TKDE 2017 (A) — Nguyen Quoc Viet Hung, Huynh Huu Viet, Nguyen Thanh Tam, Matthias Weidlich, Hongzhi Yin, Xiaofang Zhou. **Computing Crowd Consensus with Partial Agreement**. In TKDE 2017. (IF 3.438)

Conference Papers

VLDB 2019 (A^*) — Nguyen Thanh Tam, Matthias Weidlich, Hongzhi Yin, Bolong Zheng, Nguyen Quoc Viet Hung, Bela Stantic. **User Guidance for Efficient Fact Checking**. In VLDB 2019.

VLDB 2019 (A^*) — Nguyen Thanh Tam, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Nguyen Quoc Viet Hung, Bela Stantic. **From Anomaly Detection to Rumour Detection using Data Streams of Social Platforms**. In VLDB 2019.

ICDE 2018 (A^*) — Nguyen Quoc Viet Hung, Kai Zheng, Matthias Weidlich, Bolong Zheng, Hongzhi Yin, Nguyen Thanh Tam, Bela Stantic. **What-if Analysis with Conflicting Goals: Recommending Data Ranges for Exploration**. In ICDE 2018.

IJCAI 2017 (A^*) — Nguyen Thanh Tam, Matthias Weidlich, Duong Chi Thang, Hongzhi Yin, Nguyen Quoc Viet Hung. **Retaining data from streams of social platforms with minimal regret**. In IJCAI 2017.

ICDE 2015 (A^*) — Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Chau Vinh Tuan, Tri Kurniawan Wijaya, Zoltan Miklos, Karl Aberer, Avigdor Gal and Matthias Weidlich. **SMART: A tool for analyzing and reconciling schema matching networks**. In ICDE 2015.

ICDE 2015 (A^*) — Nguyen Thanh Tam, Nguyen Quoc Viet Hung, Matthias Weidlich and Karl Aberer. **Result Selection and Summarization for Web Table Search**. In ICDE 2015.

DEXA 2015 (A) — Duong Chi Thang, Nguyen Thanh Tam, Nguyen Quoc Viet Hung and Karl Aberer. **An Evaluation of Diversification Techniques**. In DEXA 2015.

DASFAA 2015 (A) — Nguyen Quoc Viet Hung, Do Son Thanh, Nguyen Thanh Tam, Karl Aberer. **Tag-based Paper Retrieval: Minimizing User Effort with Diversity Awareness**. In DASFAA 2015.

DASFAA 2014 (A) Nguyen Quoc Viet Hung, Do Son Thanh, Nguyen Thanh Tam, Karl Aberer. **Privacy-Preserving Schema Reuse**. In DASFAA 2014.

ICDE 2014 (A^*) Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Zoltan Miklos, Karl Aberer, Avigdor Gal and Matthias Weidlich, **Pay-as-you-go Reconciliation in Schema Matching Networks**. In ICDE 2014.

WISE 2013 (A) Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Lam Ngoc Tran, Karl Aberer, **An Evaluation of Aggregation Techniques in Crowdsourcing**, In WISE 2013.

SIGIR 2013 (A^*) Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Lam Ngoc Tran, Karl Aberer, **A Benchmark for Aggregation Techniques in Crowdsourcing**, In SIGIR 2013.

DASFAA 2013 (A) Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Zoltan Miklos, Karl Aberer, **On Leveraging Crowdsourcing Techniques for Schema Matching Networks** , In DASFAA 2013 (Best Student Paper Award)

**Ce document a été imprimé au Centre d'impression EPFL,
imprimerie climatiquement neutre, certifiée myClimate.**