STATNet: Spatial-temporal attention in the traffic prediction

Seyed Mohamad Moghadas[1], Amin Gheibi[1], and Alexander Alahi[2]

[1]Mathematics and Computer Science Department, Amirkabir University of Technology, Iran

[2]École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

SHORT SUMMARY

Recent traffic flow prediction methods are lacking abilities to determine predictive features. Thus, they will propagate the error in the next timestamps. In this paper, first, we assess the role of spatial and temporal features on the traffic speed prediction task. Secondly, we propose an attention-based architecture to effectively leverage both cues. Our model mainly consists of two major building blocks to capture the spatial-temporal features in the data and dynamically calculate the attentive features. More specifically, the first block sequentially applies temporal convolution to produce time-based features and then employs graph convolution to capture spatial features. The second component determines the attention between spatial and temporal features. The combination of the component's output will be calculated to generate the final prediction results. Experiments on two real-world large-scale road network traffic datasets (i.e., METR-LA and PEMS-BAY) demonstrate that the proposed STATNet (spatial-temporal attention traffic network) model outperforms the state-of-the-art baselines such as graph-wavenet and STGCN (spatial-temporal graph convolutional networks).

Keywords: Spatial-temporal Attention, Graph Neural Networks, Traffic speed forecasting

1. INTRODUCTION

Accurate real-time prediction of traffic flow is very helpful for authorities to develop Intelligent Transportation Systems (ITS) which can substantially improve the traveler's experience. Traffic Speed Prediction (TSP), as a branch of traffic state prediction, has been verified to be useful for many traffic applications such as route guidance, flow control, and navigation(C. Zhang, James, & Liu, 2019). In a nutshell, predicting traffic speed, which could affect fuel consumption, environmental pollution, and difficulties in implementing public transportation management, is a typical problem of spatial-temporal forecasting task. Traffic data are recorded in certain timestamps (e.g., every 5 minutes) and at fixed locations so-called nodes. According to the connectivity of the locations by roads, which will be modeled by graphs, these data are correlated dynamically in the spatial and the temporal domain. The factors like non-linear dependencies in spatial and temporal data and external conditions such as weather have issued challenges to systems to be able to forecast accurately. To deal with high-dimensional spatial-temporal datasets, Deep learning methods seem to have promising performances, i.e., convolutional neural network (CNN) is employed to capture the spatial features of grid-based data. Also, graph convolutional neural networks are used for modeling spatial correlation of graph-based data.

The spatial-temporal prediction task is a very important research topic in machine learning. Most of the traditional methods like ARIMA(Li, Yu, Shahabi, & Liu, 2018) and SVM (Drucker, Burges,

Kaufman, Smola, & Vapnik, 1997) only consider temporal information into account. It is challenging to integrate complex spatial dependencies into prediction methods. The ConvLSTM (Shi, Yao, Tian, & Jiang, 2016) model is an extension of fully-connected LSTM(Graves, Mohamed, & Hinton, 2013), which combines CNN and RNN to model spatial and temporal correlations respectively. It utilizes CNN's powerful capability in spatial information extraction. ST-ResNet(J. Zhang, Zheng, & Qi, 2017) is a CNN-based deep residual network for citywide crowd flows prediction, which shows the power of deep residual CNN on modeling spatial-temporal grid data. ST-3DNet (Guo, Lin, Li, Chen, & Wan, 2019) introduces 3D convolutions into this area, which can effectively extract features from both the spatial and temporal dimensions. It uses two components to model the local temporal patterns and the long-term temporal patterns respectively. All of the methods above are designed for spatial-temporal grid data. Recently, researchers try to utilize graph convolution methods to model the spatial correlations in spatial-temporal network data. DCRNN (Li et al., 2018) introduces graph convolutional networks into spatial-temporal network data prediction, which uses a diffusion graph convolution network to describe the information diffusion process in spatial-temporal networks. It uses RNN to model temporal correlations like ConvLSTM. STGCN (B. Yu, Yin, & Zhu, 2018) uses CNN to model temporal correlations. ASTGCN (Guo et al. 2019a) uses two stacked attention layers to capture the dynamics of spatial dependencies and temporal correlations. Graph WaveNet (Wu, Pan, Long, Jiang, & Zhang, 2019) designs a self-adaptive matrix to take the variations of the influence between nodes and their neighbors into account. It uses dilated casual convolutions to model the temporal correlations to increase the receptive field exponentially. However, these models captured predictive spatial features by graph convolution layer, they use divergent methods to extract temporal features.

Recently, attention-based models like transformers have resulted in significant success in generating accurate outputs in both computer vision and NLP tasks. VIT(Dosovitskiy, Beyer, Kolesnikov, & Weissenborn, 2021) and GPT-3 ((Brown et al., 2020)) are great examples of successful transformer models in the vision and language domain respectively. There are some Instances of research applying the attention concept in the spatial-temporal domain such as (Guo, Lin, Feng, Song, & Wan, 2019) which extend the concept to the spatial attention and the temporal attention then utilize these blocks in their architecture. However, in their research, the temporal features have been selected in a handcrafted manner. Moreover, in addition to the latter research, (C. Zhang et al., 2019) stack the spatial and temporal attention blocks. Moreover, (Shang, Chen, & Bi, 2021) applied the recurrent graph neural network, GRU model as backbone, to capture the spatial and temporal features hidden in the network, however, the highly computation-consuming of the model, around 40 million parameters, motivate us to develop a lightweight model with same performance. We encourage to decrease the number of parameters alongside with the training time. Consequently, mentioned research, which have tried just a few configurations, motivates us to discover the effective position of the spatial and temporal attention blocks relate to each other and as a result utilize them in a performant architecture, meanwhile, constrain the number of parameters.

Overall, the contributions of our research are as follows:

- We propose a novel architecture that adaptively attends to spatial and temporal attention.
- We evaluate the impact of spatial and temporal attention separately on the traffic prediction task to afford the insight into which one is more impactful.
- Comprehensive experiments are performed on two real-world large-scale road network traffic datasets and the results show that our model consistently outperforms all the baseline methods.

In the next section, the problem will be formulated and consequently, the proposed model will be explained. In section 3, the experiments and relevant discussion will be presented.

2. METHODOLOGY

In this section, the problem will be officially defined and the building blocks of architecture will be explained. Inspired by (Vaswani et al., 2017), we proposed Spatial and Temporal attention modules processing input while addressing spatial and temporal correlation as mentioned in the latest section. Consequently, to fuse these features, they will be concatenated. We merge this encoding with the output of the GCN-based model and will produce the final result.

Problem definition

A graph is represented by $G = (V, E)$ where V is the set of nodes (traffic stations) and E is the set of edges. The adjacency matrix derived from a graph is denoted by $A \in R^{N \times N}$. If $v_i, v_j \in V$ and $(v_i, v_j) \in E$ then A_{ij} is one, otherwise it is zero. At each timestamp t, graph G contains a dynamic feature matrix (e.g., traffic speed) $X^t \in R^{N \times D}$ where N is the number of nodes and D is the number of traffic features. In the literature, the feature matrix is used interchangeably with graph signals. Given a graph G and its historical S time-stamps graph signals, our problem is to learn a function f that can forecast its next T step graph signals. The above mapping is shown as follows:

$$\left[X^{(t-S):t}, G\right] \xrightarrow{f} X^{(t+1):(t+T)} \tag{1}$$

Where $X^{(t-S):t} \in R^{N \times D \times S}$ and $X^{(t+1):(t+T)} \in R^{N \times D \times T}$.

Proposed model

To model spatial and temporal correlations and involve them effectively into prediction, we design the STATNet model. Another architecture criteria would be designing a lightweight model.As shown in Figure1, the model contains two major components:

1. Traffic prediction engine which in this paper is inspired by Graph Wavenet (Wu et al., 2019). Overall, two class of models, CNN-based and RNN-based, could be selected. However, to get rid of the recurrent nature of RNNs, based on that they are not able to be paralleled and also need a huge amount of computation resources, a CNN-based model has been picked.
2. Attention module which comprises spatial and temporal attention blocks alongside with transformer-related layers.

In the following sections, the details of each component will be detailed. First, the spatial and temporal attention will be defined then the main GCN-based model will be explained.

2.2.1 Spatial Attention

To capture the impact of spatial features, the spatial attention concept has been used. Based on the proposed spatial attention (Guo, Lin, Feng, et al., 2019), we use initial spatial attention which means it has been employed on just input features. Suppose that N, D, T would be the number of nodes, number of feature dimension and the size of time window respectively. The spatial attention representation mapping would be defined as:

$$\begin{aligned} SA &= V_{sa} \cdot \sigma((X^T W_1) W_2 (W_3 X)^T) + b_{sa} \\ SA' &= Softmax(SA) \end{aligned} \tag{2}$$

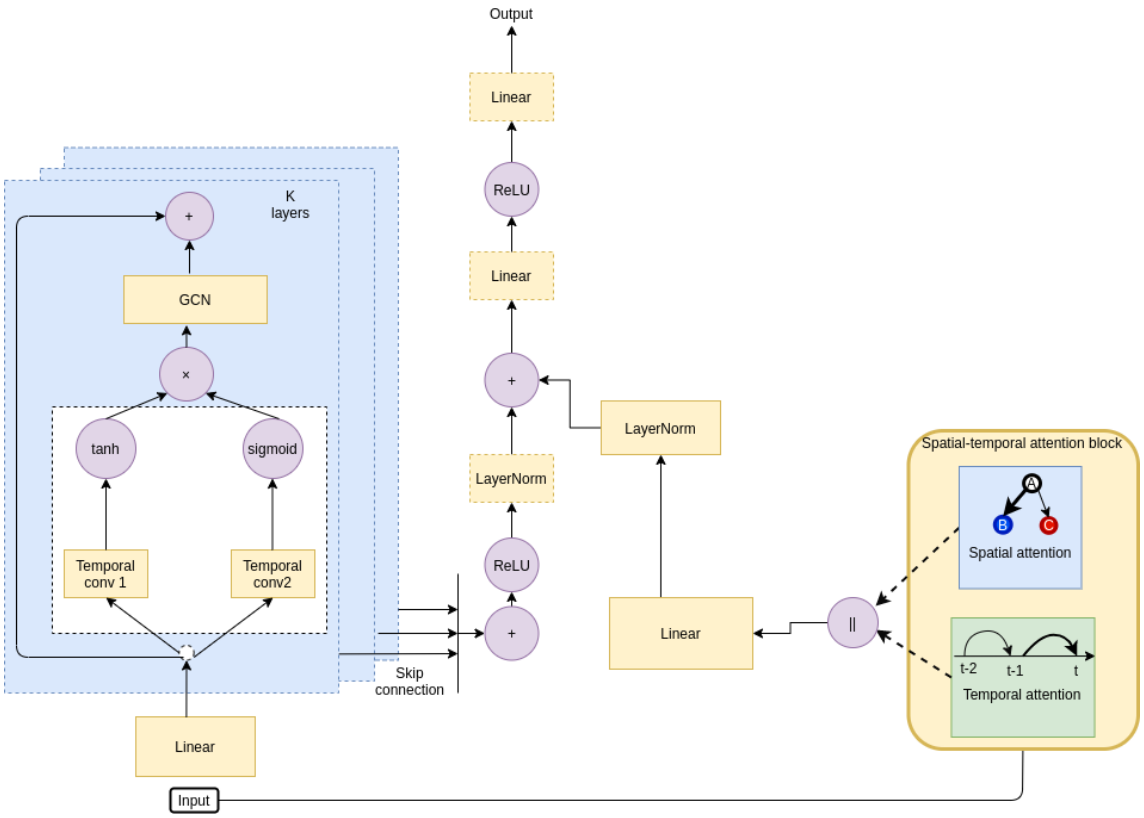

Figure 1: Proposed model. Types of spatial-temporal attention configuration will explained at next paragraph

where W_i, V_{sa} are trainable parameters, $V_{sa} \in R^{N \times N}, W_1 \in R^T$, $W_2 \in R^{D \times T}$, $W_3 \in R^D$, and b_s is the bias and σ is Sigmoid function. Semantically, it indicates the amount of attention between each pair of nodes at each timestamp.

2.2.2 Temporal Attention

Correspondingly, the temporal attention concept could be employed to capture the temporal correlation in different timestamps within the traffic flow.

$$\begin{aligned} TA &= V_{ta} \cdot \sigma((X^T U_1) U_2 (U_3 X)^T) + b_{ta} \\ TA' &= Softmax(TA) \end{aligned} \tag{3}$$

where U_i and V_{ta} are trainable parameters, $V_{ta} \in R^{T \times T}, U_1 \in R^N$, $U_2 \in R^{D \times N}$, $U_3 \in R^D$, and b_e is the bias and σ is Sigmoid function. This block would be responsible of capturing temporal patterns in the different timestamps. Despite of the previous works, there is no limitations in the temporal features. According to these notations, the V_{ta} and V_{sa} could be sort of an embedding in the temporal and spatial domains.

2.2.3 Spatial-temporal attention

However in recent years, most of the state-of-the-art spatial-temporal attention models like (Guo, Lin, Feng, et al., 2019) use these blocks in the cascade model, in order to determine the importance of each block and finding a appropriate combination, we employ them in parallel and sequential

architectures. On the other hand, in most well-known models in this field, e.g., (C. Zhang et al., 2019), the spatial attention is placed amongst two temporal attention layers claimed to prevent overfitting. So to address the question we try three architecture shown in Fig. 2. The overall architecture which adaptively calculates spatial attention and temporal attention weights is called STATNet. Based on our experiments, spatial features are more important because of containing larger weight after the end of learning. Due to being unified with the classical transformer (Vaswani et al., 2017), the layer normalization (Layernorm) has been used. Our experiments have shown its regularization effect causing performance improvement.

2.2.4 Graph Convolution Layer

Graph convolution is an essential operation to extract a node's features given its structural information. (Berg, Kipf, & Welling, 2017) proposed the first approximation of Chebyshev spectral filter (Defferrard, Bresson, & Vandergheynst, 2016). From a spatial-based perspective, it smoothed a node's signal by aggregating and transforming its neighborhood information. One of the advantages of their method is that its filter is localized in spatial domain, and it could be scaled for high-dimensional data. However, the studies have shown that it has generalization issues for unseen nodes in the same graph or in an entirely different graph (S. Zhang, Tong, Xu, & Maciejewski, 2019).

2.2.5 Temporal convolution

We adopt the dilated causal convolution (F. Yu & Koltun, 2015) as our temporal convolution layer (TCN) to capture a node's temporal trends. Dilated causal convolution networks allow an exponentially large receptive field by increasing the layer depth. As opposed to RNN-based approaches, dilated casual convolution networks can handle long-range sequences properly in a non-recursive manner, which facilitates parallel computation and prevents the gradient explosion problem. The dilated causal convolution preserves the temporal causal order by padding zeros to the inputs so that predictions made on the current time step only involve historical information. As a special case of standard 1D convolution, the dilated causal convolution operation slides over inputs by skipping values with a certain step, as illustrated by Figure 3. Mathematically, given a 1D sequence input $x \in R^k$ and a filter $f \in R^k$, the dilated causal convolution operation of x with f at step t is represented as follows:

$$x \star f(t) = \sum_{s=0}^{K-1} f(s)x(t - d \times s) \tag{4}$$

where d is the dilation factor that controls the skipping distance.

3. RESULTS AND DISCUSSION

We evaluate our STATNet model on two public traffic network datasets, METR-LA and PEMS-BAY released by (Li et al., 2018). METR-LA records four months of statistics on traffic speed on 207 sensors on the highways of Los Angeles County. PEMS-BAY contains six months of traffic speed information data pre-processing procedures as in(Li et al., 2018). The readings of the sensors are aggregated into 5-minutes horizons. The adjacency matrix of the nodes is constructed by road network distance with a thresholded Gaussian kernel. Z-score normalization is applied to

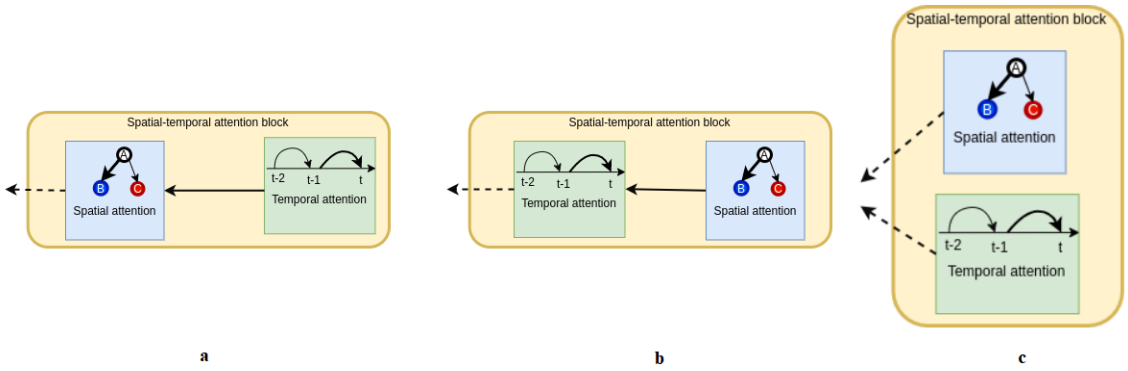

Figure 2: Various configurations of the spatial-temporal attention block:sequentially (a,b) and jointly (c)

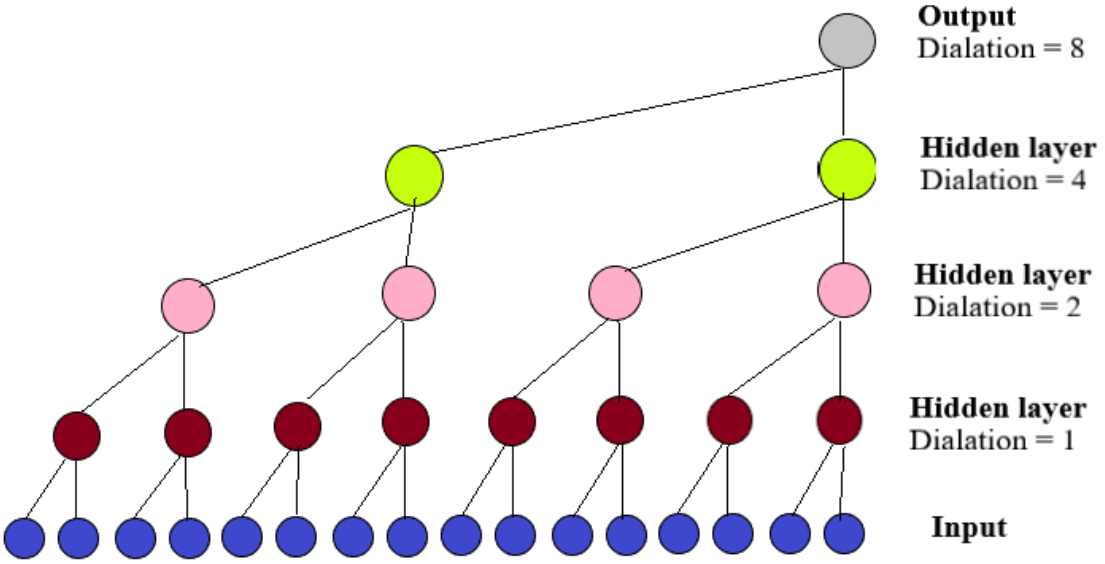

Figure 3: Temporal convolution model

inputs. The datasets are split in chronological order with 70% for training, 10% for validation, and 20% for testing. Detailed dataset statistics are provided in Table 1.

Table 1: Summary statistics of METR-LA and PEMS-BAY.

Data	Nodes	Edges	Time Steps
METR-LA	207	1515	34272
PEMS-BAY	325	2369	52116

Baselines

We compare STATNet in different settings against the following models.

- **ARIMA(Li et al., 2018):** Auto-Regressive Integrated Moving Average model with Kalman filter.
- **FC-LSTM(Li et al., 2018):** Recurrent neural network with fully connected LSTM hidden units.
- **WaveNet(van den Oord et al., 2016):** A convolution network architecture for sequence data.
- **DCRNN(Li et al., 2018):** Diffusion convolution recurrent neural network combines graph convolution networks with recurrent neural networks in an encoder-decoder manner.
- **GGRU(M. Zhang & Chen, 2018):** Graph gated recurrent unit network Recurrent-based approaches. GGRU uses attention mechanisms in graph convolution.
- **STGCN(B. Yu et al., 2018):** Spatial-temporal graph convolution network which combines graph convolution with 1D convolution.
- **Graph Wavenet(Guo, Lin, Feng, et al., 2019):** Capturing temporal features with convolution and applied GCN on the self-adaptive adjacent matrix.
- **GTS(Shang et al., 2021):** Graph for Time Series, which applied GGRU on the Gamble-sampled input graph.

Experimental Setups

Our experiments are conducted under a cluster environment with two NVIDIA V100 PCIe 32 GB GPUs (2×7TFLOPS) cards. To cover the input sequence length, we use eight layers of Graph WaveNet with a sequence of dilation factors 1, 2, 1, 2, 1, 2, 1, 2. We use Equation 4 as our graph convolution layer with a diffusion step K = 2. We randomly initialize node embeddings by a uniform distribution with a size of 10. We train our model using Adam optimizer with an initial learning rate of 0.001. Dropout with p=0.35 is applied to the outputs of the graph convolution layer. The evaluation metrics we select include mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). Missing values are excluded from the dataset (including training and testing).

Experimental results

Table 2 compares the performance of STATnet and baseline models for 15 minutes, 30 minutes, and 60 minutes ahead prediction on METR-LA and PEMS-BAY datasets. STATNet obtains superior results on both datasets. It outperforms temporal models including ARIMA, FCLSTM,

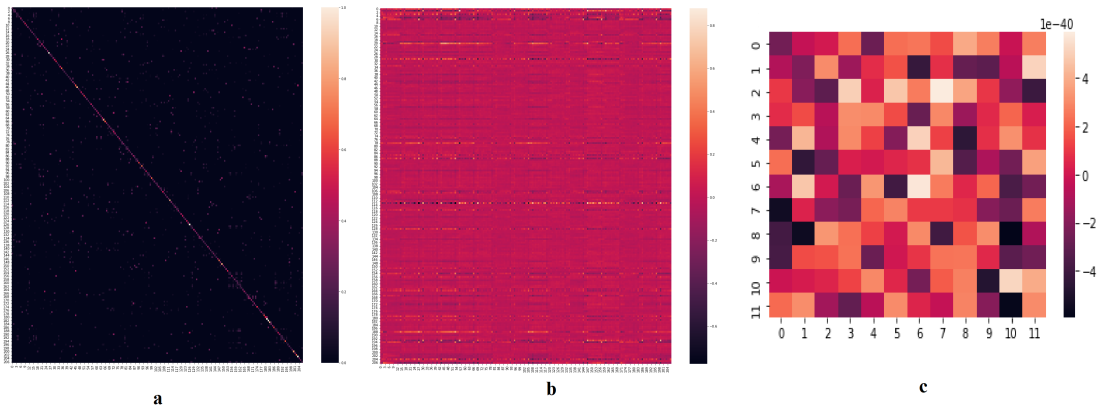

Figure 4: a. Adjacent matrix heatmap of static input graph for METR-LA dataset b. Spatial attention layer weight heatmap. c. Temporal attention layer weight heatmap

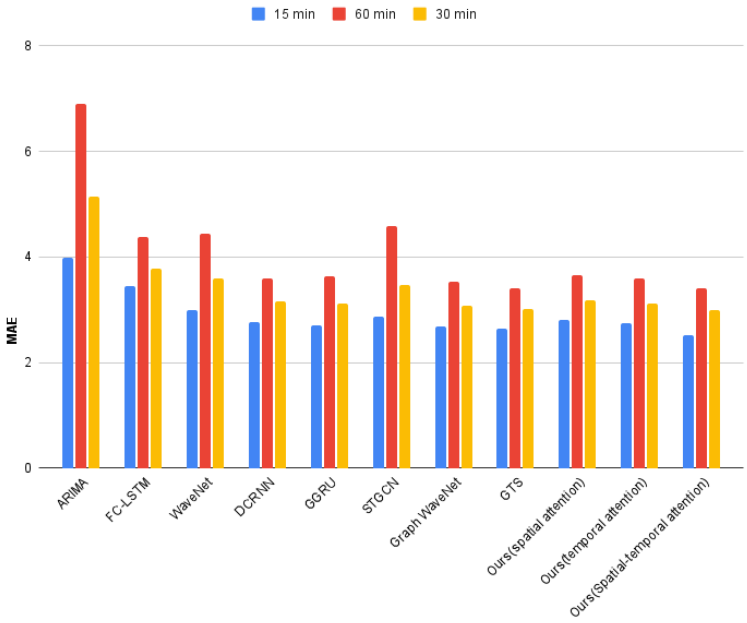

Figure 5: Performance of baselines compare to our method(in METR-LA dataset)

and WaveNet by a large margin. Compared to other spatial-temporal models, STATNet surpasses the previous convolution-based approach STGCN significantly and outperforms GRNN (Graph recurrent neural network) approaches DCRNN and GGRU at the same time. In respect of the second-best model GTS as shown in Table 2, STATNet adaptively involves predictive features by finding a suitable combination of spatial and temporal attention. According to Table 2, in the merely spatial attention or temporal attention setting STATNet does not outperform the baselines. Moreover, as shown in Fig. 4, the temporal attention heatmap approves the non-linear dependency between different timestamps as well as non-linear spatial dependency. It was obtained for the 12 consecutive timestamps in the 5 minute periods. Conceivably, the spatial attention heatmap follows the neighborhood patterns. In other words, it visualizes learned representation by the GCN. Finally, the comparison of the performance of our proposed method and the other baselines shown in Fig. 5 for the horizons of 15min, 30min, and 60min.

Table 2: The performance comparison of our model on the METR-LA and PEMS-BAY dataset.

Dataset	Model	15 min			30 min			60 min		
		MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
METR-LA	ARIMA(Li et al., 2018)	3.99	8.21	9.60%	5.15	10.45	12.70%	6.9	13.23	17.40%
	FC-LSTM(Li et al., 2018)	3.44	6.3	9.60%	3.77	7.23	10.90%	4.37	8.69	13.20%
	WaveNet(van den Oord et al., 2016)	2.99	5.89	8.04%	3.59	7.28	10.25%	4.45	8.93	13.62%
	DCRNN(Li et al., 2018)	2.77	5.38	7.30%	3.15	6.45	8.80%	3.6	7.6	10.50%
	GGRU(M. Zhang & Chen, 2018)	2.71	5.24	6.99%	3.12	6.36	8.56%	3.64	7.65	10.62%
	STGCN(B. Yu et al., 2018)	2.88	5.74	7.62%	3.47	7.24	9.57%	4.59	9.4	12.70%
	Graph WaveNet(Guo, Lin, Feng, et al., 2019)	2.69	5.15	6.90%	3.07	6.22	8.37%	3.53	7.37	10.01%
	GTS(Shang et al., 2021)	2.64	**4.95**	6.80%	3.01	**5.85**	8.20%	3.41	**6.74**	9.90%
	Ours(spatial attention)	2.8	5.16	7.29%	3.17	6.05	8.56%	3.66	7.35	10.35%
	Ours(temporal attention)	2.74	5.15	7.15%	3.11	6.03	8.54%	3.59	7.3	10.35%
	Ours(Spatial-temporal attention)	**2.51**	5.13	**6.26%**	**2.99**	6.01	**8.08%**	**3.4**	7.1	**9.57%**
PEMS-BAY	ARIMA(Li et al., 2018)	1.62	3.3	3.50%	2.33	4.76	5.40%	3.38	6.5	8.30%
	FC-LSTM(Li et al., 2018)	2.05	4.19	4.80%	2.2	4.55	5.20%	2.37	4.96	5.70%
	WaveNet(van den Oord et al., 2016)	1.39	3.01	2.91%	1.83	4.21	4.16%	2.35	5.43	5.87%
	DCRNN(Li et al., 2018)	1.38	2.95	2.90%	1.74	3.97	3.90%	2.07	4.74	4.90%
	GGRU(M. Zhang & Chen, 2018)	-	-	-	-	-	-	-	-	-
	STGCN(B. Yu et al., 2018)	1.36	2.96	2.90%	1.81	4.27	4.17%	2.49	5.69	5.79%
	Graph WaveNet(Guo, Lin, Feng, et al., 2019)	1.3	2.74	2.73%	1.63	3.7	3.67%	1.95	4.52	4.63%
	GTS(Shang et al., 2021)	1.32	2.62	2.80%	1.64	3.41	3.60%	1.91	3.97	4.40%
	Ours(spatial attention)	1.35	2.66	2.88%	1.68	3.46	3.75%	2.02	4.22	4.78%
	Ours(temporal attention)	1.32	2.65	2.82%	1.65	3.47	3.74%	1.98	4.28	4.78%
	Ours(Spatial-temporal attention)	**1.19**	**2.64**	**2.45%**	**1.44**	**3.45**	**3.13%**	**1.68**	**4.03**	**3.86%**

Table 3: Result of experiments on the various spatial-temporal block configurations

Dataset	Model	15 min			30 min			60 min		
		MAE	RMSE	MAPE	MAE	RMSE	MAPE	MAE	RMSE	MAPE
METR-LA	Ours(temporal-spatial)	2.75	5.19	7.25%	3.17	6.07	8.46%	3.67	7.29	10.3%
	Ours(spatial-temporal)	2.67	5.16	7.13%	3.10	6.02	8.44%	3.57	7.22	10.26%
	Ours(Spatial-temporal-joint)	**2.51**	**5.13**	**6.26%**	**2.99**	**6.01**	**8.08%**	**3.4**	**7.1**	**9.57%**

Another criteria was minimizing number of parameters. So, according to the Table 4, the number of parameters for our model is significantly less than (Shang et al., 2021).

As mentioned in last section, the spatial-temporal attention module has been tested by joint and sequential configuration. According to table 3, the joint architecture has led to the best result. One reason would be representing the spatial and temporal features simultaneously would be more predictive than treating them separately. The similar method has been applied in the Multimodal NLP (Gomez, Gibert, Gomez, & Karatzas, 2019), which concatenated the textual and image features. Interestingly, in the temporal-spatial setup, the results were even worse then single-feature loss. To compare our model with GTS (Shang et al., 2021), in term of computational resource consumption, our model is much more efficient because our model's quantity of trainable parameters are

Table 4: The number of trainable parameters comparison of our model with the baselines

Model	# of trainable params
Graph WaveNet	997,528
GTS	38,377,237
Ours	**3,698,131**

substantially lower, as the training time, while in RMSE metric, our model could not outperform GTS (Shang et al., 2021) model.

4. CONCLUSIONS

In this paper, we have presented a novel model for spatial-temporal attention graph modeling. Our model has captured spatial-temporal dependencies efficiently by combining graph convolution with convolution and involving adaptive spatial-temporal attention. We have proposed an effective method to learn hidden spatial-temporal attention automatically from the data. Based on the observations for the absence of spatial or temporal attention, it could be concluded that spatial and temporal attention, both are crucial for predictive graph representation. This will open up a new direction in spatial-temporal graphs to evaluate the models akin to a transformer. For instance, the self-attention could be extended based on spatial and temporal features. On two public traffic network datasets, STATNet achieves state-of-the-art results while the number of parameters would be much less. The code is available : https://github.com/moghadas76/statnet

ACKNOWLEDGMENT

We gratefully acknowledge the support of VITA lab for providing the infrastructure. We would like to thank the anonymous reviewers for their detailed comments and constructive feedback.

REFERENCES

Berg, R. v. d., Kipf, T. N., & Welling, M. (2017). Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., & Neelakantan, A. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Defferrard, M., Bresson, X., & Vandergheynst, P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, *29*, 3844–3852.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., & Weissenborn, D. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Drucker, H., Burges, C., Kaufman, L., Smola, A., & Vapnik, V. (1997, 01). Support vector regression machines. *Advances in neural information processing systems*, *28*, 779-784.

Gomez, R., Gibert, J., Gomez, L., & Karatzas, D. (2019). Exploring hate speech detection

in multimodal publications. *arXiv preprint arXiv:1910.03814*.

Graves, A., Mohamed, A.-r., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In *2013 ieee international conference on acoustics, speech and signal processing* (pp. 6645–6649).

Guo, S., Lin, Y., Feng, N., Song, C., & Wan, H. (2019, Jul.). Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(01), 922-929. Retrieved from https://ojs.aaai.org/index.php/AAAI/article/view/3881 doi: 10.1609/aaai.v33i01.3301922

Guo, S., Lin, Y., Li, S., Chen, Z., & Wan, H. (2019). Deep spatial–temporal 3d convolutional neural networks for traffic data forecasting. *IEEE Transactions on Intelligent Transportation Systems*, *20*(10), 3913–3926.

Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2018). Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*.

Shang, C., Chen, J., & Bi, J. (2021). Discrete graph structure learning for forecasting multiple time series. In *International conference on learning representations*. Retrieved from https://openreview.net/forum?id=WEHSlH5mOk

Shi, Y., Yao, K., Tian, L., & Jiang, D. (2016). Deep lstm based feature mapping for query classification. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 1501–1511).

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wu, Z., Pan, S., Long, G., Jiang, J., & Zhang, C. (2019). Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121*.

Yu, B., Yin, H., & Zhu, Z. (2018, Jul). Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. Retrieved from http://dx.doi.org/10.24963/ijcai.2018/505 doi: 10.24963/ijcai.2018/505

Yu, F., & Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Zhang, C., James, J., & Liu, Y. (2019). Spatial-temporal graph attention networks: A deep learning approach for traffic forecasting. *IEEE Access*, *7*, 166246–166256.

Zhang, J., Zheng, Y., & Qi, D. (2017). Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Thirty-first aaai conference on artificial intelligence*.

Zhang, M., & Chen, Y. (2018). Link prediction based on graph neural networks. *arXiv preprint arXiv:1802.09691*.

Zhang, S., Tong, H., Xu, J., & Maciejewski, R. (2019, November). Graph convolutional networks: a comprehensive review. *Computational Social Networks*, *6*(1). Retrieved from https://doi.org/10.1186/s40649-019-0069-y doi: 10.1186/s40649-019-0069-y