MultiSPANS: A Multi-range Spatial-Temporal Transformer Network for Traffic Forecast via Structural Entropy Optimization

Dongcheng Zou
Beihang University
Beijing, China
zoudongcheng@buaa.edu.cn

Senzhang Wang*
Central South University
Changsha, Hunan, China
szwang@csu.edu.cn

Xuefeng Li
Beihang University
Beijing, China
xuefengli@buaa.edu.cn

Hao Peng*
Beihang University
Beijing, China
penghao@buaa.edu.cn

Yuandong Wang*
Tsinghua University
Beijing, China
wangyd2021@tsinghua.edu.cn

Chunyang Liu
Didi chuxing
Beijing, China
liuchunyang@didiglobal.com

Kehua Sheng
Didi chuxing
Beijing, China
shengkehua@didiglobal.com

Bo Zhang
Didi chuxing
Beijing, China
zhangbo@didiglobal.com

ABSTRACT

Traffic forecasting is a complex multivariate time-series regression task of paramount importance for traffic management and planning. However, existing approaches often struggle to model complex multi-range dependencies using local spatiotemporal features and road network hierarchical knowledge. To address this, we propose MultiSPANS. First, considering that an individual recording point cannot reflect critical spatiotemporal local patterns, we design multi-filter convolution modules for generating informative *ST-token* embeddings to facilitate attention computation. Then, based on *ST-token* and spatial-temporal position encoding, we employ the Transformers to capture long-range temporal and spatial dependencies. Furthermore, we introduce structural entropy theory to optimize the spatial attention mechanism. Specifically, The structural entropy minimization algorithm is used to generate optimal road network hierarchies, i.e., encoding trees. Based on this, we propose a relative structural entropy-based position encoding and a multi-head attention masking scheme based on multi-layer encoding trees. Extensive experiments demonstrate the superiority of the presented framework over several state-of-the-art methods in real-world traffic datasets, and the longer historical windows are effectively utilized. The code is available at https://github.com/SELGroup/MultiSPANS.

*Corresponding authors

WSDM '24, March 4–8, 2024, Merida, Mexico

ACM ISBN 979-8-4007-0371-3/24/03…$15.00
https://doi.org/10.1145/3616855.3635820

CCS CONCEPTS

• **Information systems** → **Spatial-temporal systems**; **Traffic analysis**; • **Mathematics of computing** → **Time series analysis**.

KEYWORDS

Traffic forecast; multivariate time-series forecast; spatial-temporal data mining; structural entropy; convolution network; Transformer

ACM Reference Format:
Dongcheng Zou, Senzhang Wang, Xuefeng Li, Hao Peng, Yuandong Wang, Chunyang Liu, Kehua Sheng, and Bo Zhang. 2024. MultiSPANS: A Multi-range Spatial-Temporal Transformer Network for Traffic Forecast via Structural Entropy Optimization. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM '24), March 4–8, 2024, Merida, Mexico.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3616855.3635820

1 INTRODUCTION

Transportation is a complex real-world system that includes people, vehicles, road network sensors, and other components, with a wealth of temporal and spatial connections. As urbanization continues to advance, there is an increasing demand for more precise analysis of traffic data to improve the efficiency of transportation systems. To address the growing complexity of traffic-related tasks, deep learning approaches have been widely employed for route planning [23, 28, 37], flow prediction [10, 15, 21, 62], accident prediction [14, 39, 55], vehicle scheduling [43, 52], etc. One of the fundamental technologies for intelligent transportation is traffic state forecast, which can be considered as a multivariate time series regression task. It involves modeling temporal and spatial dependencies to predict future traffic situations (e.g., flow, speed, or occupancy) based on prior road networks, historical observations, and external traffic-related information.

Current fundamental time-series methods for traffic forecast tasks include Recurrent Neural Networks (RNNs) [2, 21, 61] and Temporal Convolutional Networks (TCNs) [26, 46, 47], while Graph

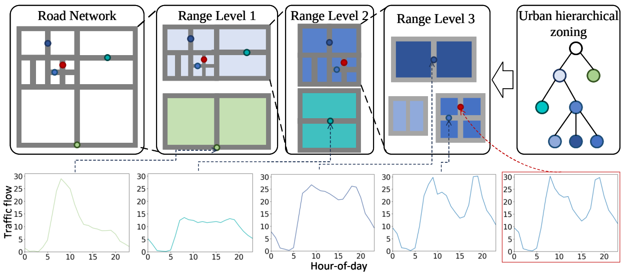

Figure 1: An toy example of the hierarchical urban zoning and its impact on traffic flow.

Neural Networks (GNNs) are commonly used to factor the spatial attributes [21, 47, 61]. These works still face challenges, such as difficulty in modeling long-range dependencies [1, 3], dealing with time-varying graphs [12], and coping with unreliable structures [46]. Recently, Transformer [38] has been widely used in spatiotemporal tasks to address existing issues. However, these frontier Transformer-based methods have two problems corresponding to time-series and graph learning. First, Transformers may not be as effective as expected in handling long time-series data [22, 56]. It is possibly because the information in discrete time points is insufficient to learn pairwise attention and model higher-order global temporality [15, 30]. Second, Transformers have difficulty in directly utilizing the graph structure. Mainstream approaches include fusing GNN and Transformer output [49, 62] or obtaining simple attention masks/encoding [15, 53] from networks. These structure learning mechanisms for Transformers are designed without theoretical guidance and may ignore the rich structural information.

To address both issues in spatial-temporal Transformers, we aim to improve the network to capture rich spatiotemporal dependencies from multiple ranges. Motivated by patching techniques in the visual [6] Transformer, we aim to extract and aggregate multi-frequency local spatiotemporal signals to obtain more representational *ST-tokens* as the basis for effective attention computation. Further, we expect Transformer to focus on urban functional zoning impact on traffic state (i.e., greater correlation in the same section). As shown in Fig. 1, functional zoning is naturally hierarchical, reflected in the road network, hard to predefine, and highly correlated with traffic states, e.g., roads in the same high-level community have similar flow characteristics. Therefore, we introduce the structural entropy theory to measure the uncertainty of the road network and obtain the hierarchical zoning unsupervisedly and adaptively. Specifically, we propose MultiSPANS: a **Multi**-range **S**patiotemporal **P**rediction **A**ttention **N**etwork with **S**tructural entropy optimization. First, we design a lightweight multi-filter convolution module comprising temporal filters graph filters for *ST-tokens* with extensive local information. Then, we organize the network by interleaving multiple temporal and spatial Transformers to enhance the model's fitting capability toward complex traffic data. Moreover, an innovative hierarchical graph perception mechanism based on structural entropy is presented. Structural entropy [17] can measure the complexity of a road network and guide the optimal graph hierarchical abstraction by creating the encoding tree [19]. According to the structural entropy and multi-level encoding tree, we devised hierarchical correlation scores to identify the nodes' position in the hierarchical community, and multi-level attention masks to learn the relevance at different structural levels separately on each attention head. The main contributions are outlined below:

- A novel and effective spatial-temporal Transformer network, MultiSPANS, is proposed for a more accurate and versatile traffic state forecast, which addresses current issues. Experiments validate that our method achieves new SOTA in real-world road network datasets.
- A practical and pluggable spatial-temporal convolutional module is proposed to obtain informative *ST-tokens* for Transformers in spatiotemporal tasks. It can embed longer historical windows with high computational efficiency to enhance the model's ability to handle long time series.
- The structural entropy theory is first exploited to optimize the spatial attention mechanism, which mines the hierarchical structure of the road networks. Visualization study shows that our method can intuitively model multi-range spatial dependencies and discover more relative patterns.

2 PRELIMINARIES

2.1 Problem Definition

The C-channel (speed, flow, occupancy, etc.) traffic state signal collected by the n-th sensor at the moment t (i.e., atomic data point) can be represented by the vector $x_{n,t} \in \mathbb{R}^C$. The traffic state feature in a time window of width T (starting from moment t) for a road network with N sensor nodes can be represented as:

$$X_{[t,t+T]} = \begin{bmatrix} \begin{bmatrix} x_{1,t+1} \\ x_{2,t+1} \\ \cdots \\ x_{N,t+1} \end{bmatrix} & \begin{bmatrix} x_{1,t+2} \\ x_{2,t+2} \\ \cdots \\ x_{N,t+2} \end{bmatrix} & \cdots & \begin{bmatrix} x_{1,t+T} \\ x_{2,t+T} \\ \cdots \\ x_{N,t+T} \end{bmatrix} \end{bmatrix} \in \mathbb{R}^{T \times N \times C}. \tag{1}$$

The traffic state forecasting problem aims to predict future traffic states according to historical observations, prior structure, and additional information, which can be formalized as:

$$\hat{X}_{[t,t+T']} = f_\theta \left(X_{[t-T,t]}, A_{t-T}, G \right), \tag{2}$$

where f_θ is model with parameter θ, $\hat{X}_{[t,t+T']}$ is the predicted time window of width T', and A_{t-T} is the addition information of the historical window. G denotes the topology structure, which can be road network maps or dynamic graph sequences.

2.2 Graphs and Structural Entropy

Let $G = \{V, E\}$ denote a graph, where V is the set of N vertices [1] and $E \subseteq V \times V$ is the edge set. $A \in \mathbb{R}^{N \times N}$ denotes the adjacency matrix of G, where A_{ij} is referred to as the weight of the edge from vertex i to vertex j. The degree of vertex $v_i \in V$ is defined as $d(v_i) = \sum_j A_{ij}$, and $D = \text{diag}(d(v_1), d(v_2), \ldots, d(v_N))$ refers to the degree matrix. Recent research by Li and Pan [17] has systematically presented the structural information theory, aiming to measure the uncertainty and information embedded in graphs and obtain the informative

[1]Vertices are defined in the graph and nodes are in the tree.

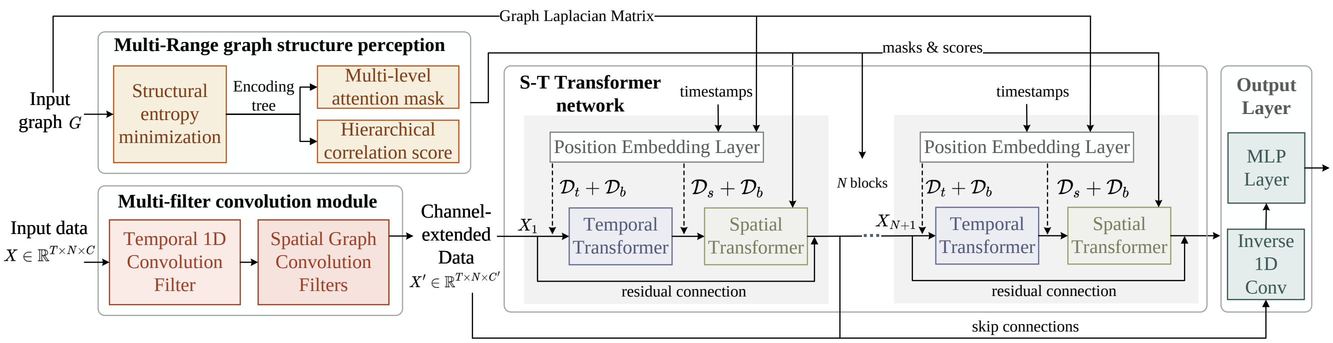

Figure 2: The overall architecture of MultiSPANS.

hierarchical structures for graph compression. The theory mainly consists of two parts: Encoding Tree and Structural Entropy.

Encoding Tree An encoding tree is a hierarchy that encodes and compresses graphs. For the graph $G = \{V, E\}$, the encoding tree T rooted at node λ is defined with the following properties: 1) For each node α in T, its associated vertex (e.g., the physical node in graph G) set is defined as $\mathcal{T}_\alpha \subseteq V$. 2) For each node α, its parent node is denoted as α^- and its i-th children node is denoted as $\alpha^{\langle i \rangle}$ ordered from left to right as i increases. 3) For each non-leaf node α with N children, all vertex subset $\mathcal{T}_{\alpha^{\langle i \rangle}}$ satisfy $\mathcal{T}_\alpha = \bigcup_{i=1}^{N} \mathcal{T}_{\alpha^{\langle i \rangle}}$ and $\bigcap_{i=1}^{N} \mathcal{T}_{\alpha^{\langle i \rangle}} = \varnothing$. Thus, the encoding tree abstracts and encodes the graph into a hierarchical community structure.

Structural Entropy Structural entropy is determined by the encoding tree and the graph together, which can be formulated as follows:

$$H^{\mathrm{T}}(G) = \sum_{\alpha \in \mathrm{T}, \alpha \neq \lambda} H^{\mathrm{T}}(G; \alpha) = - \sum_{\alpha \in \mathrm{T}, \alpha \neq \lambda} \frac{g_\alpha}{vol(G)} \log_2 \frac{\mathcal{V}_\alpha}{\mathcal{V}_{\alpha^-}}, \quad (3)$$

where g_α is the sum weights of edges from the vertices outside $\mathcal{T}_\alpha$ to those inside $\mathcal{T}_\alpha$. $vol(G)$ is the sum degree of all vertices in G, and $\mathcal{V}_\alpha$ is the sum degree in $\mathcal{T}_\alpha$. The encoding tree that minimizes the graph's structural entropy compresses the most knowledge. Therefore, taking the total information in the graph as constant, it is optimal to represent the essential graph hierarchical structure.

3 PROPOSED METHOD

3.1 Overall Architecture

Fig. 2 depicts the comprehensive architecture, encompassing three primary sub-modules: the multi-filter convolutional (MFCL) module, the spatial-temporal (ST) Transformers, and the hierarchical graph perception mechanism. Firstly, we employ the MFCL module to obtain *ST-tokens*, including multiple 1D filters to enhance temporal signals at diverse frequencies and multi-hop graph convolutional filter to aggregate neighborhood signals (§ 3.2). Next, we model complex dependencies with the Transformer network, consisting of a stack of ST encoders with residual connections. Each ST encoder comprises two sequentially arranged temporal and spatial Transformers. (§ 3.3). The skip connections of each ST encoder are summed and fed into an output layer with a transposed 1D convolutional layer(§ 3.4). Furthermore, we propose a hierarchical graph structure perception mechanism for spatial attention based on structural entropy optimization to exploit the rich information embedded in road networks. It abstracts the graph into a hierarchy (i.e., encoding tree), based on which we present multi-level attention masks to regularize spatial attention and hierarchical correlation scores as relative position encoding (§ 3.3.3).

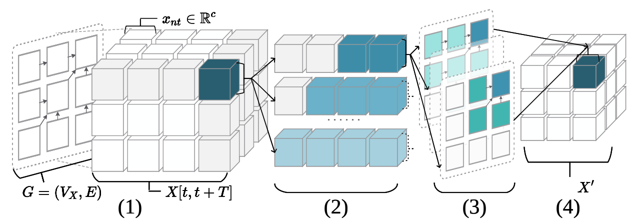

Figure 3: An illustration of the workflow of multi-filter convolution module. (1) 3D spatiotemporal data with the T-step time window and a predefined graph G. Each atomic data point has c-channel attribute; (2) Multiple temporal convolutional filters are employed to extract diverse short-range time patterns; (3) Graph convolutional filters are added for neighborhood aggregation that facilitates the local spatial pattern; (4) Processed data enjoy more extensive channels c_t.

3.2 Multi-filter Convolution Module

The multi-filter convolution (MFCL) module aims to expand the dimensionality and enrich the information of token embeddings while incorporating more intricate local spatiotemporal features and patterns. We employ two specific designs: multi-frequency temporal convolution filters and multi-hop graph convolution filters. Fig. 3 illustrates the data structure and workflow of this module.

Temporal Convolution Filter Recognizing the inherent periodicity of the traffic system, we employ a set of standard 1D filters with various sizes to extract short-range temporal features at multiple frequencies. Suppose there are m filters with sizes $k_1, k_2, \cdots, k_m$, the temporal convolution operation with c-channel input and c_t-channel output at time t can be formulated as follows:

$$x'_t = ||_{j=1}^{m} \sum_{i=1}^{c} \sum_{l=1}^{k_j} W^{(j)}[l, i] X[i, t + l - k_j], \quad (4)$$

where $X \in \mathbb{R}^{T\times c}$ is input time series, $x'_t \in \mathbb{R}^{c_t}$ denotes the output at step t, and $W^{(j)} \in \mathbb{R}^{k_j \times c \times (c_t/m)}$ is the kernel matrix of j-th filter (where c_t must be divisible by m). $[\cdot]$ is the index operation, and $||$ is the concatenation operation along the channel dimension. By concatenating the multiple filters' results, the channel of temporal data is extended to c_t. Since all filters are expected to produce sequences of a uniform length, we padded the sequence to $T \leftarrow T + k_j - 1$ in length by duplicating the first and last point of the sequence before feeding into the j-th filter. The size and number of convolution filters can be customized for different tasks to accommodate larger historical windows, and the uniform stride of temporal filters can be enlarged to compress the sequence. Our basic implementation selects four filters with size 1×1, 1×2, 1×3, and 1×6, often corresponding to intervals of 5, 10, 15, and 30 minutes.

Graph Convolution Filter To extract the short-range spatial pattern of the traffic state that propagates on the road network, multi-hop graph convolution filters are adopted to fuse the node feature within the neighborhood. Denoting the 1-hop adjacency matrix of the graph as A, the h-hop graph convolution operation with c_t-channel can be formulated as:

$$x'_n = ||_{j=0}^{h} \sum_{i=1}^{N} \hat{A}^j[n,i]X[I], \hat{A} = D^{-1}(A+I). \tag{5}$$

The kernel matrix $\hat{A}$ is derived by adding the self-loop matrix I to A and normalizing it with the degree matrix D. $X \in \mathbb{R}^{N\times c_t}$ denotes the node features with c_t-channel at the moment t after temporal filtering, and x'_n is the output of the n-th node. $\hat{A}^j$ refers to the j-th power of $\hat{A}$, which acts as the multi-hop graph convolution filter that aggregation messages from j-hop neighbors. Finally, all the outputs are concatenated into a vector of dimension $d = (h+1) \cdot c_t$, allowing each data point to aggregate the multi-hop neighborhood locality of the road network. Our method aggregates the neighborhood representation of each hop independently and has fewer trainable parameters than other similar designs [1, 46].

3.3 Spatial-Temporal Transformer Network

3.3.1 Position Embedding Layer. First, to integrate the spatial node position within the spatial Transformer, we utilize the Laplacian graph matrix to encode the road network topology into static representations [8]. Specifically, We compute the node eigenvectors of G via $U^T \Lambda U = I - D^{-1/2}AD^{-1/2}$, where U and Λ correspond to eigenvalues and eigenvectors. A linear projection $W \in \mathbb{R}^{k\times d}$ is applied on k smallest non-trivial eigenvectors to generate the spatial embedding $\mathcal{D}_s \in n \times d$. Second, we employ the *Sinusoidal* position encoding $\mathcal{D}_t \in t\times d$ based on the original Transformer [38] design to incorporate temporal sequential information. In addition, for continuous time-series datasets, the position of the current batch within the entire dataset needs to be considered. We perform one-hot encoding on the day-of-week and hour-of-day timestamps of the data batch and map them into $\mathcal{D}_b \in t \times d$ to account for cross-batch periodicity. Finally, $H + \mathcal{D}_t + \mathcal{D}_b$ and $H + \mathcal{D}_s + \mathcal{D}_b$ are fed into the spatial and temporal Transformer, respectively. Here, $H \in \mathbb{R}^{T\times N\times d}$ is the hidden output state of the previous module due to sequential arrangement.

3.3.2 Spatial-Temporal Transformer. In order to model global spatiotemporal dependencies on global road networks and historical windows, we employ the unified transformer module with h-head

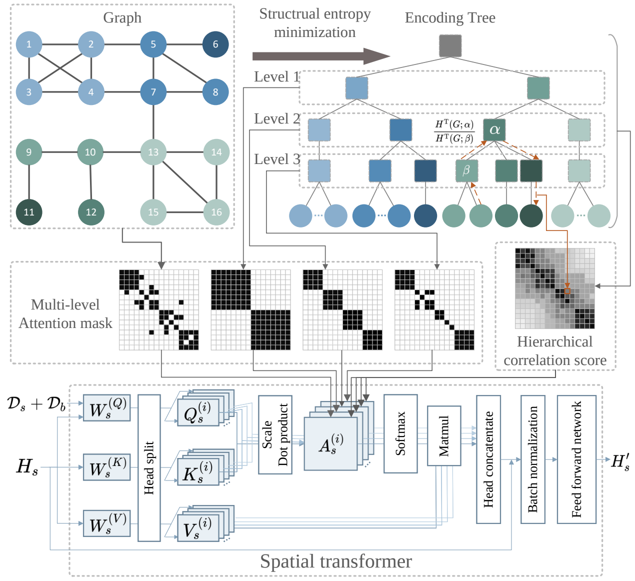

Figure 4: An illustration depicting the hierarchical graph perception mechanism and the spatial Transformer.

attention, which can be formulated as:

$$Q = W_Q^{(i)}(H + \mathcal{D} + \mathcal{D}_b), K = W_Q^{(i)}H, V = W_V^{(i)}H, \tag{6}$$

$$A^{(i)} = ((Q^{(i)} \cdot K^{(i)})^T + S/\sqrt{d}) \odot M^{(i)}, \tag{7}$$

$$H' = \text{Norm}(\text{RelU}(W_{ffn} \cdot ||_{i=1}^{h}(\text{softmax}(A^{(i)})V^{(i)}))). \tag{8}$$

For the i-th attention head, H is the spatiotemporal input (as Fig. 3 shows) and $W_Q^{(i)}$,$W_K^{(i)}$, and $W_V^{(i)}$ are learnable linear projection. $A^{(i)}$ is the attention matrix, S is an addition similarity matrix (also denoted as relative position encoding), and $M^{(i)}$ is the attention mask, which is Hadamard product ($\odot$) with $A^{(i)}$. The final outputs H' of all heads are concatenated into d-channel and further fed into a channel-mixing feed-forward layer, where W_{ffn} is the parameter of the feed-forward network and Norm is the batch normalization. The structure of the Temporal and Spatial Transformer is basically the same, but there are still the following differences: 1. the position encodings D are differently obtained. As described in § 3.3.1, D in Temporal Transformer is D_t, while it is D_s in Spatial Transformer; 2. Temporal Transformer only models the relationship between time points, with all spatial locations share a set of projection parameters, while the opposite is the case for spatial Transformer; 3. We design a unique relative location encoding S and a multi-head attention mask M for spatial attention in § 3.3.3.

3.3.3 Multi-Range Graph Structure Perception. The urban fabric has a natural hierarchy due to its functional division (e.g. residential, commercial, etc.), which can be reflected by the road network structure and influence the traffic state. Structural entropy and encoding tree theory are innovatively introduced to mine higher-order knowledge from the road network and incorporate it into the self-attention mechanism. Firstly, we apply the structural entropy minimization algorithm to obtain an optimal encoding tree, which serves as a hierarchical abstraction of the road network. Secondly, we use the hierarchy to model the low-rank relationship within the network and propose multi-level attention masks. Finally, we propose the hierarchical correlation score based on the relative position of physical (leaf) nodes on the encoding tree, which reveals the road network's underlying structure and node positions.

Road Network Abstraction Drawing inspiration from the principle of structural entropy minimization [17], we introduce a heuristic algorithm and corresponding tree operators (i.e., the combination operator and merge operator) from deDoc [19] to compute the optimal encoding tree of road network G to obtain a hierarchical zoning structure. First, we initial a flat encoding tree (with only one level where all leaf nodes are direct descendants of the root node). For each iteration, the node pair and operator that maximize the reduction of structural entropy are selected and conducted in a greedy manner. In the end, the algorithm terminates when the structural entropy ceases to decrease continuously, resulting in the final optimal encoding tree denoted as T^*.

Multi-level Attention Mask The number of levels in an encoding tree generally depends on the size of the graph and its structural complexity and can be determined adaptively during the optimization. Each level of the encoding tree corresponds to a partition of the graph node-set, representing the road network potential zoning at a specific spatial scale. Given $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ on the l-th level of the optimal encoding tree T^* and $\mathcal{T}_\lambda = \bigcup_{i=1}^{l} \mathcal{T}_{\alpha_i}$, we can acquire the mask matrix $M^{(l)} \in \{-INF, 1\}^{N\times N}$ that satisfied

$$m_l[i,j] = \begin{cases} 1 & \text{if } \exists \alpha_m \in \{\alpha_1,\dots,\alpha_l\}, v_i \in \mathcal{T}_{\alpha_m}, v_j \in \mathcal{T}_{\alpha_m} \\ -INF & \text{else} \end{cases}, \tag{9}$$

where $m_l[i,j]$ denotes the element in the i-th row and j-th column of M_l. For an L-level encoding tree, we can obtain $L-1$ mask matrices with diverse granularity from every level except for the leaf level. In addition, we introduce an additional adjacency matrix as the L-th mask to capture edge-level local relations with the minimum range. The L masks are applied to the H attention heads (ensuring $H > L$) to capture dependencies within different ranges, whereas the extra $H - L$ attention heads are unmasked to model the wide global attention.

Hierarchical Correlation Score The multi-level attention mask can leverage low-rank constraints on multi-head spatial attention within structural levels but may ignore vertical cross-level relations in hierarchies. Therefore, we design a relative position encoding to identify vertices in graphs based on the optimal encoding trees. Specifically, we define the relative structural entropy based on the encoding tree T. For nodes α and β that have an inheritance relationship, the structural entropy of α relative to β is defined as $H^{\mathrm{T}}_{rel}(G;\alpha|\beta)$=$H^{\mathrm{T}}(G;\alpha)/H^{\mathrm{T}}(G;\alpha)$. It reflects the relative complexity and informativeness between the vertices and sub-structures of the graph G. Then, assuming two leaf nodes α_i and α_j of the encoding tree share the lowest common ancestor θ, the structural entropy of α_i relative to α_j can be defined as follows:

$$H^{\mathrm{T}}_{rel}(G;\alpha_j|\alpha_i) = H^{\mathrm{T}}_{rel}(G;\theta|\alpha_i) + H^{\mathrm{T}}_{rel}(G;\alpha_j|\theta) = \sum_{\beta,\mathcal{T}_{\alpha_i}\subseteq\mathcal{T}_\beta\subset\mathcal{T}_\theta} H^{\mathrm{T}}(G;\beta^-|\beta) + \sum_{\beta,\mathcal{T}_\theta\supset\mathcal{T}_\beta\supseteq\mathcal{T}_{\alpha_j}} H^{\mathrm{T}}(G;\beta|\beta^-). \tag{10}$$

From another perspective, we view the encoding tree as a graph and add up the relative structural entropy of the connected nodes on the shortest directed path between two leaf nodes α_j, α_i to obtain the final relative structural entropy, based on which can we generate the hierarchical correlation matrix satisfying that $S_{hier}[i,j] = H^{\mathrm{T}}_{rel}(G;\alpha_j|\alpha_i)$ where $\mathcal{T}_{\alpha_i} = v_i$ and $\mathcal{T}_{\alpha_j} = v_j$. The hierarchical correlation score S_{hier} enables attention to prioritize more intricate structures while preserving the hierarchical information of the road network. In conclusion, in order to improve the mechanisms of spatial attention, the road network is first abstracted into an encoding tree via the structural entropy minimization algorithm. Then each level i of the encoding tree (and the adjacency matrix) is constructed as an attention mask $M^{(i)}$ that operates on a specific attention head. Furthermore, a hierarchical correlation score S_{hier} derived from relative structural entropy is employed as a prior score and is added to attention matrices. The modified Spatial Transformer module is depicted in Fig.4.

3.4 Output Layer

After collecting all intermediate outputs of the ST encoder blocks and the multi-filter convolution filter with the skip connections, they are summed into $H_o \in \mathbb{R}^{T\times N\times D}$ and fed into a deconvolution decoder and an MLP decoder. The deconvolution smoothly extends the predicted sequence if the dimension of the hidden states length T is inconsistent with the multi-step predicted length T'. The MLP projects the output's channel dimension and sequence length to the desired shape and obtains the final prediction $H_o \in \mathbb{R}^{T'\times N\times C_o}$.

4 RESULTS AND ANALYSIS

4.1 Experimental Settings

Implementation. All experiments were performed on the NVIDIA GeForce 3090 with 24GB of memory. The model was trained by Adam optimizer [25] with a mean absolute loss (MAE) for 50 epochs, employing the learning rate $1e-2$ and batch size 32. The datasets were partitioned into training, validation, and test sets with a ratio of $6:2:2$. The model with the best validation performance was selected for testing. For a fair comparison, we uniformly configured the number of ST layers as $k = 3$, the hidden dimension as $d = 64$, and the heads number in self-attention as $h = 8$ for all baselines.

Datasets. We conduct experiments on traffic dataset **PEMSD4** [11] and **PEMSD8** [11]. Both include flow, speed, and occupancy information, with an interval of 5 minutes. We use all channels as input and select one as the output, based on which we derive four subsets: PEMSD4-speed, PEMSD4-flow, PEMSD8-speed, and PEMSD8-flow.

Evaluation Metrics. Three metrics, mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE), are used for evaluation. Additionally, the average error of output steps was reported to evaluate comprehensively.

Baselines. We compare MultiSPANS against the following baseline methods of four types. **Traditional Methods**: Models that apply traditional machine learning methods, including Support Vector Regression (SVR) [7] and Vector Auto Regression(VAR) [27]; **Deep Learning Methods**: Methods that apply deep approaches excluding GNN or attention, including AutoEncoder(AE) [29] and LSTM [13]; **Advanced Methods**: Model specialized for spatiotemporal traffic data with a subtle combination of TCN/RNN and GNN,

Table 1: Experiment Results of the average 12-step forecast. The best results are bolded, and the runner-up results are underlined. *Our* indicates the performance of our purposed MultiSPANS. *Imp.* denotes the improvement of our method over the SOTA method.

Dataset	Metrics	VAR	SVR	AE	LSTM	TGCN	DCRNN	STGCN	MTGNN	GWNet	ASTGCN	STTN	GMAN	Our	Imp.
PEMSD4-flow	MAE	24.98	27.45	24.59	23.80	22.88	22.63	21.60	<u>19.29</u>	19.53	19.56	19.49	19.35	**19.07**	1.10%
	MAPE	18.24	19.83	16.48	15.78	14.52	13.97	14.68	13.54	<u>13.41</u>	13.91	13.78	13.57	**13.29**	0.90%
	RMSE	38.91	40.74	37.63	35.92	34.41	34.70	34.76	31.82	31.95	32.03	31.87	<u>31.62</u>	**30.46**	3.30%
PEMSD8-flow	MAE	27.46	32.83	20.48	19.48	18.61	18.42	17.92	15.47	<u>15.09</u>	15.92	15.63	15.34	**14.68**	2.72%
	MAPE	16.82	15.97	13.43	14.85	11.47	11.10	11.36	10.16	**9.63**	10.66	10.46	10.22	<u>9.79</u>	-
	RMSE	45.01	43.95	35.19	33.27	27.95	28.14	27.34	24.93	<u>24.84</u>	25.37	25.26	25.13	**23.87**	4.31%
PEMSD4-speed	MAE	3.29	3.15	2.35	2.58	1.94	1.70	1.80	1.67	<u>1.66</u>	1.80	1.72	1.74	**1.61**	3.01%
	MAPE	5.90	5.77	4.79	4.17	3.77	3.60	3.57	3.48	<u>3.45</u>	3.94	3.68	3.64	**3.39**	1.74%
	RMSE	5.72	6.02	4.98	5.07	4.18	3.95	3.02	3.76	<u>3.71</u>	3.97	3.72	3.72	**3.66**	1.35%
PEMSD8-speed	MAE	3.14	3.60	2.13	2.35	1.73	1.51	1.55	1.47	<u>1.42</u>	1.59	1.54	1.49	**1.36**	4.23%
	MAPE	6.39	6.48	5.04	4.96	3.42	3.26	3.28	<u>2.95</u>	3.06	3.62	3.61	3.41	**2.84**	3.73%
	RMSE	6.83	6.13	5.35	5.29	3.67	3.64	3.50	3.49	3.56	3.73	3.90	<u>3.43</u>	**3.26**	4.96%

including TGCN [61], STGCN [54], MTGNN [46], and GWNET [47]; **Transformer-based Methods**: Methods using attention to capture both spatial and temporal dependencies, including ASTGCN [10], STTN [49], and GMAN [62]; Implementation of the baselines comes from the Libcity[2] [40] benchmark and is adapted to our settings.

4.2 Experimental Result

4.2.1 Comparison with baselines. A comprehensive comparison between the MultiSPANS and the baselines is conducted, and the results are reported in Table 1. Evidently, all deep learning-based approaches outperform traditional ones in traffic forecast, and further improvements can be achieved by introducing and improving GNN or Transformer for better spatiality. We observed that Transformer-based methods generally perform better than GNN-RNN (e.g., STGCN and DCRNN) methods due to their stronger ability to capture global and dynamic dependencies. However, MTGNN and GWNET, based on TCN and GNN, show competitive performance and even outperform Transformer-based methods. This may be attributed to their adaptive graph structure learning modules. The MultiSPANS exhibits remarkable performance superiority over baseline methods across all datasets. Compared to the SOTAs, MultiSPANS achieves an average improvement of 2.57%, 2.16%, and 3.78% for MAE, MAPE, and RMSE, respectively. Particularly, MultiSPANS achieves the most significant improvement on PEMSD8-speed, which delivers impressive results of MAE 1.36, MAPE 2.84, and RMSE 3.26, corresponding to the improvements of 4.23%, 3.73%, and 4.96%, respectively. Additionally, We found that MultiSPANS performs exceptionally well in RMSE, with 23.87 in PESMD8-flow and 30.46 in PESMD4-flow, which may be attributed to the smoothing and denoising impact of the MFCL module and transposed convolutional output layer.

4.2.2 Long time-series modelling experiments. In this subsection, we explore the ability of MultiSPANS to model larger historical time windows and choose a convolution-based (i.e., STGCN) and a transformer-based approach (i.e., STTN) for comparison.

We adopt the stride of $1, 3, 4$ for the $12, 36, 48$ steps historical window for a uniform 12- length hidden state in MultiSPANS. In Table 2, $I48, I36, I12$ represent using historical windows of length $48, 36, 12$. MultiSPANS^1 denotes the MultiSPANS with original settings, while MultiSPANS^2 denotes it with 8 temporal filters of size $[1, 2, 3, 4, 6, 12, 18, 24]$. *Paras.* reports models' total parameter numbers. *Time* reports the average time cost of an epoch. The best results are bolded. As can be observed in Table. 2, expanding the history window can improve the performance in most cases, but the extra time and space cost varies among the methods. In particular, the improvement in STTN is disproportionate to its incremental time consumption, mainly due to the increasing computation in dynamic spatial attention on more time patches. Meanwhile, STGCN improved significantly with longer historical windows, possibly owing to the notable increase in learnable parameters, which also require larger memory. However, the proposed MultiSPANS can compress the hidden temporal dimension by tuning the stride of the temporal convolutional filters, thus allowing longer-range history windows to be exploited for improved forecast results at trivial additional cost. Furthermore, extending the number of temporal filters to extract more frequencies of short-range patterns can considerably improve the performance of MultiSPANS to model long-range with a MAE of 18.85, MAPE of 13.17, and RMSE of 30.18.

Table 2: Results with longer windows on PESMD4-flow.

Model	MAE	MAPE	RMSE	Paras.	Time
MultiSPANS$^2_{-I48}$	**18.85**	13.19	**30.18**	332.3K	269.15s
MultiSPANS$^2_{-I36}$	18.93	**13.17**	30.25	332.3K	269.48s
MultiSPANS$^1_{-I48}$	19.06	13.21	30.33	332.0K	266.39s
MultiSPANS$^1_{-I36}$	19.01	13.24	30.28	332.0K	266.19s
MultiSPANS$^1_{-I12}$	19.07	13.29	30.46	332.0K	259.46s
STTN$_{-I48}$	19.31	13.55	31.74	699.8K	931.18s
STTN$_{-I36}$	19.40	13.62	31.69	700.1K	693.41s
STTN$_{-I12}$	19.49	13.78	31.87	700.2K	178.64s
STGCN$_{-I48}$	20.97	14.42	33.35	1565.5K	62.72s
STGCN$_{-I36}$	21.31	14.45	33.44	1172.3K	43.95s
STGCN$_{-I12}$	21.60	14.68	34.76	385.9K	15.56s

[2]https://libcity.ai/#/

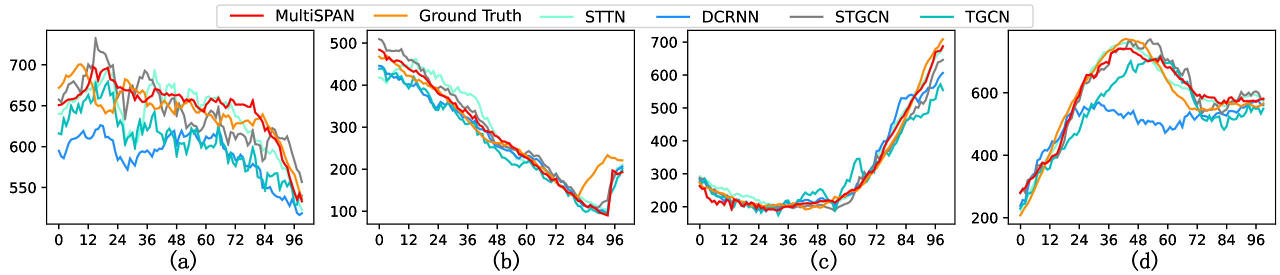

Figure 5: Forecast results for different periods at the same location. We visualized the traffic flow over 100 consecutive time steps using the average results of multiple 12-step forecasts and the ground truth.

4.3 Ablation Studies

In this subsection, we conduct an ablation study on the PEMSD4-flow dataset by removing specific modules to evaluate their effectiveness, and results are presented in Table 3.

To thoroughly evaluate the multi-filter convolution (MFCL) module, we perform three experiments: (1) removing the temporal filter (w\o TF), (2) removing the spatial filter (w\o SF), and (3) replacing the MFCL module with a linear layer(w\o MFCL). It is evident that the improvement of MFCL is dramatic, reaching a surprising 5.24%. Meanwhile, the temporal filter is more effective than the spatial, contributing a 1.98% improvement compared to 1.49%. This observation highlights the necessity of the multi-filter convolution module to extract local patterns for the long-range attention mechanism.

To evaluate the effectiveness of the hierarchical graph perception mechanism, we design experiments to remove or modify its components. Specifically, we (1) remove the multi-level attention mask(w\o mask), (2) remove the hierarchical correlation score(w\o score), (3) remove the whole mechanism(w\o mask), and (4) use the Infomap [33] algorithm, a minimum entropy-based hierarchical community detection method, to construct the multi-level mask(w Infomap). The results show that both the multi-level attention mask and hierarchical correlation score significantly improve the model's performance, contributing to a 2.33% and a 1.52% improvement, respectively. And the total improvement of the proposed mechanism amounts to 4.55%, compared to the vanilla attention. This suggests that our approach efficiently incorporates topological knowledge into the multi-headed attention, effectively capturing spatial dependencies. Furthermore, our structural entropy-based method outperforms the Infomap-based method, indicating that structural entropy optimization is more suitable for road network hierarchy abstraction. Overall, these analyses demonstrate that our design effectively supports multi-range spatio-temporal modeling for traffic.

Table 3: Effects of different MultiSPANS components.

Model	MAE	MAPE	RMSE	Imp.
MultiSPANS	19.07	13.29	30.46	-
w\o TF	19.49	13.56	30.98	1.98%
w\o SF	19.42	13.44	30.92	1.49%
w\o MFCL	20.04	14.13	31.78	5.24%
w\o mask	19.48	13.79	30.79	2.33%
w\o bias	19.32	13.56	30.83	1.52%
w\o both	19.73	14.3	31.25	4.55%
w Infomap	19.43	13.58	30.89	1.83%

4.4 Case Studies

4.4.1 Temporal Dependency Study. Figure 5 presents the average prediction of methods in different periods at the same location, along with the corresponding ground truth. Specifically, we display the flow prediction of DCRNN, STTN, STGCN, TCN, MultiSPANS, and the ground truth starting from time steps 72 (a), 432 (b), 792 (c), and 1152 (d) of node 101 in the PEMSD4. In (b), (c), and (d), our model's results are smoother and less sensitive to anomalies in comparison. This can be attributed to the denoising effect of the incorporated multi-filter convolution module and temporal deconvolution decoder. And overall, our model fits the ground truth better, matching trends (b,c) and effectively modeling specific temporal patterns (a,d), indicating its efficiency in temporal modeling.

4.4.2 Spatial Dependency Study. We also illustrate the spatial attention map captured by MultiSPANS in Fig. 6. As shown in Fig. (a), attention is modeled globally without masks, and most nodes rely heavily on a few key nodes in the road network. Fig. (b) shows the discrete attention matrix when using the adjacency matrix as a mask. Both attention-modeling approaches drastically lose sight of the complex semantics of the road network. Meanwhile, as shown in Fig. (a)~(h), the multi-level attention we designed can capture different range dependencies at each attention head separately. The fusion of the attention map provided by the hierarchical graph perception mechanism (Fig. (c)) shows that our approach is able to model richer spatiality than vanilla attention. To interpret the plausibility of the attention of our method, we further analyze temporal patterns among closely related nodes. Specifically, we selected the three nodes with the strongest relevance to 197 points based on multi-level attention (Fig. (i)) and vanilla attention (Fig. (j)) and visualized their corresponding local flow in Fig.(k) and Fig.(l), respectively. While if no multi-level constraints are added, long-range relationships can be captured (e.g., nodes 246 and 197), but the overall similarity is not pronounced.

4.4.3 Hyperparameter Analysis. Fig. 7 evaluates two hyperparameters on PEMSD4-flow, i.e., the temporal filter number k_1 and hops of the spatial filter k_2. Appropriate k_1 and k_2 do promote model performance in terms of extracting extensive local patterns and avoiding excessive noise. Meanwhile, they generally remain at a

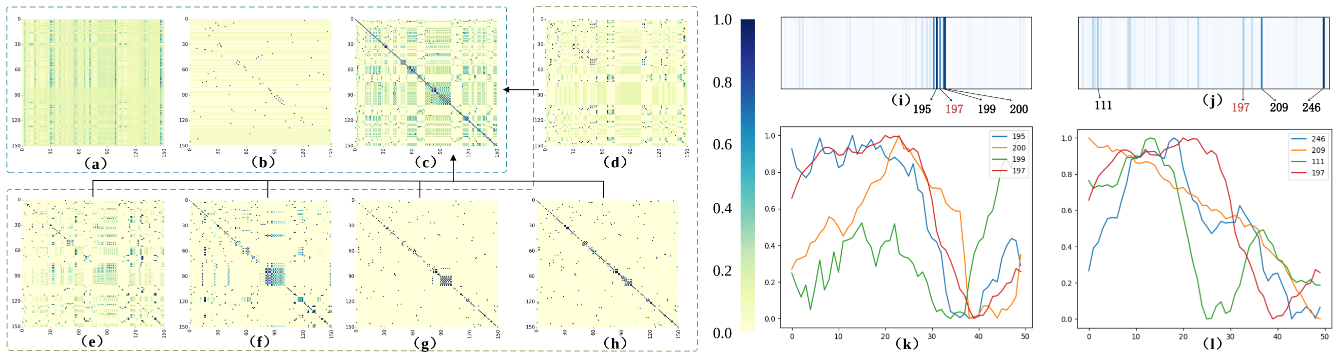

Figure 6: The heatmaps of attention score. The results of the 100th-250th nodes are shown. (a): The attention map without masks; (b): The attention map masked by adjacency matrix; (c): The average attention map from all heads in MultiSPANS; (d)~ (h): The multi-head attention maps with hierarchical graph perception mechanism from coarse to fine granularity; (i)~(j): The attention heat map between the 197th node and other nodes with vanilla attention and our multi-range method, respectively; (k)~(l): Current traffic flow at the 197th node and the top 3 relevant nodes based on vanilla attention and our method, respectively.

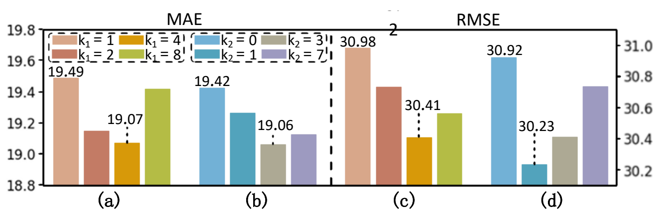

Figure 7: Influence of hyperparameters. Figure (a), (b) shows the influence of k_1, and Figure(b), (d) shows that of k_2.

high level and exhibit relative stability, indicating that our method is not sensitive to the hyperparameters. Further, Noting that even with $k_1 = 1$ or $k_2 = 0$ (i.e. with only one temporal filter or not exhibiting spatial neighborhood), MultiSPANS still achieves RMSEs of 30.98 and 30.92, respectively, exceeding most existing models.

5 RELATED WORK

Deep traffic forecast Deep traffic forecast is a spatiotemporal regression task involving GNN, RNN, TCN, and Transformer [41] etc. Learning spatiality with GNN and predicting with RNN is a typical paradigm [4, 5, 21, 48, 61]. Meanwhile, deep convolutional approaches of stacking GNN and TCN modules have also proved effective, which ameliorates the localization problem [9, 9, 26, 47, 47, 54]. To further improve the capabilities, some work aims to utilize traffic-related attributes, like hour-of-day and day-of-week etc [10, 36, 51], and some adopt graph structure learning for high-quality and task-relevant road network structures [34, 42, 46, 47, 60]. Recently, many studies [22, 30, 44, 63] have endorsed Transformers in long time series, despite some deficiencies [56] such as poor information in single tokens [30]. Therefore, advanced work [10, 12, 15, 32, 49, 62] is keen to model both temporal and spatial dependencies with Transformers. For example, ASTGNN [12] propose a dynamic tri-multi-head self-attention, and STTN [49] incorporates GCNs and spatial Transformers with the gated-fusion. PDformer [15] adopts geographic and semantic spatial masks on attention heads.

Structural entropy application. To evaluate the quality and informativeness of the graph structure, many works [17, 31, 33] are presented to extend the Shannon entropy [35] to structural data. Among which, structural information theory [17], as a de-facto solution to measure information in graphs, was first applied in network security [16, 20, 24] and bioinformatics [18, 19, 57], etc. Recently, a wave of work has been aimed at applying structural entropy to cutting-edge machine-learning areas. Some work has attempted to improve GNNs by structural entropy, i.e., selecting optimal hyperparameters [50], learning graph structures [64], or designing pooling frameworks [45]. Some work combines structural information with reinforcement learning to optimize role [58] and state [59] abstraction, with promising results achieved.

6 CONCLUSION

We address multi-range spatial modeling from the structural entropy perspective and propose a novel Transformer-based traffic forecast framework. Consisting of a multi-filter convolution module, road network abstraction, and graph perception mechanism, MultiSPANS succeeds in spatiotemporal tokenizing, discovering road network hierarchy, and poses the multi-level constraint on Transformers. Experiments show that MultiSPANS achieves excellent performance, and demonstrate the effectiveness of proposed modules. In the future, we plan to focus on applying structural entropy-guided attention mechanisms to graph and spatial data and analyze the Transformer's interpretability from the hierarchical network analysis perspective.

ACKNOWLEDGMENTS

This work is supported by National Key R&D Program of China through grant 2021YFB1714800, NSFC through grants 62322202 and 62206148, Beijing Natural Science Foundation through grant 4222030, S&T Program of Hebei through grant 20310101D, China Postdoctoral Science Foundation through grant 2022M711811, CCF-DiDi GAIA Collaborative Research Funds for Young Scholars, and the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Sami Abu-El-Haija, Bryan Perozzi, Amol Kapoor, Nazanin Alipourfard, Kristina Lerman, Hrayr Harutyunyan, Greg Ver Steeg, and Aram Galstyan. 2019. Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In *international conference on machine learning*. PMLR, 21–29.

[2] Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. 2020. Adaptive Graph Convolutional Recurrent Network for Traffic Forecasting. *Advances in Neural Information Processing Systems* 33 (2020), 17804–17815.

[3] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. 2020. Simple and deep graph convolutional networks. In *International conference on machine learning*. PMLR, 1725–1735.

[4] Weiqi Chen, Ling Chen, Yu Xie, Wei Cao, Yusong Gao, and Xiaojie Feng. 2020. Multi-Range Attentive Bicomponent Graph Convolutional Network for Traffic Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 04 (April 2020), 3529–3536. https://doi.org/10.1609/aaai.v34i04.5758

[5] Zhiyong Cui, Kristian Henrickson, Ruimin Ke, and Yinhai Wang. 2019. Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems* 21, 11 (2019), 4883–4894.

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

[7] Harris Drucker, Christopher J. C. Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. 1996. Support Vector Regression Machines. In *Advances in Neural Information Processing Systems*, Vol. 9. MIT Press.

[8] Vijay Prakash Dwivedi and Xavier Bresson. 2021. A Generalization of Transformer Networks to Graphs. https://doi.org/10.48550/arXiv.2012.09699 arXiv:2012.09699 [cs]

[9] Kan Guo, Yongli Hu, Yanfeng Sun, Sean Qian, Junbin Gao, and Baocai Yin. 2021. Hierarchical Graph Convolution Network for Traffic Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 35, 1 (May 2021), 151–159. https://doi.org/10.1609/aaai.v35i1.16088

[10] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. 2019. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 33, 01 (July 2019), 922–929. https://doi.org/10.1609/aaai.v33i01.3301922

[11] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. 2019. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 33, 01 (July 2019), 922–929. https://doi.org/10.1609/aaai.v33i01.3301922

[12] Shengnan Guo, Youfang Lin, Huaiyu Wan, Xiucheng Li, and Gao Cong. 2022. Learning Dynamics and Heterogeneity of Spatial-Temporal Graph Data for Traffic Forecasting. *IEEE Transactions on Knowledge and Data Engineering* 34, 11 (Nov. 2022), 5415–5428. https://doi.org/10.1109/TKDE.2021.3056502

[13] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9, 8 (Nov. 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[14] Chao Huang, Chuxu Zhang, Peng Dai, and Liefeng Bo. 2019. Deep dynamic fusion network for traffic accident forecasting. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 2673–2681.

[15] Jiawei Jiang, Chengkai Han, Wayne Xin Zhao, and Jingyuan Wang. 2023. *PDFormer: Propagation Delay-Aware Dynamic Long-Range Transformer for Traffic Flow Prediction*.

[16] Angsheng Li, Qifu Hu, Jun Liu, and Yicheng Pan. 2016. Resistance and security index of networks: structural information perspective of network security. *Scientific reports* 6, 1 (2016), 26810.

[17] Angsheng Li and Yicheng Pan. 2016. Structural information and dynamical complexity of networks. *IEEE Transactions on Information Theory* 62, 6 (2016), 3290–3339.

[18] Angsheng Li, Xianchen Yin, and Yicheng Pan. 2016. Three-dimensional gene map of cancer cell types: Structural entropy minimisation principle for defining tumour subtypes. *Scientific reports* 6, 1 (2016), 1–26.

[19] Angsheng Li, Xianchen Yin, Bingxiang Xu, Danyang Wang, Jimin Han, Yi Wei, Yun Deng, Ying Xiong, and Zhihua Zhang. 2018. Decoding topologically associating domains with ultra-low resolution Hi-C data by graph structural entropy. *Nature communications* 9, 1 (2018), 1–12.

[20] Angsheng Li, Xiaohui Zhang, and Yicheng Pan. 2017. Resistance maximization principle for defending networks against virus attack. *Physica A: Statistical Mechanics and its Applications* 466 (2017), 211–223.

[21] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2022. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. In *International Conference on Learning Representations*.

[22] Zhe Li, Zhongwen Rao, Lujia Pan, and Zenglin Xu. 2023. *MTS-Mixers: Multivariate Time Series Forecasting via Factorized Temporal and Channel Mixing*.

[23] Defu Lian, Yongji Wu, Yong Ge, Xing Xie, and Enhong Chen. 2020. Geography-aware sequential location recommendation. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 2009–2019.

[24] Yiwei Liu, Jiamou Liu, Zijian Zhang, Liehuang Zhu, and Angsheng Li. 2019. REM: From structural entropy to community structure deception. *Advances in Neural Information Processing Systems* 32 (2019).

[25] Ilya Loshchilov and Frank Hutter. 2023. Fixing Weight Decay Regularization in Adam. (May 2023).

[26] Bin Lu, Xiaoying Gan, Haiming Jin, Luoyi Fu, and Haisong Zhang. 2020. Spatiotemporal Adaptive Gated Graph Convolution Network for Urban Traffic Flow Forecasting. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 1025–1034. https://doi.org/10.1145/3340531.3411894

[27] Zheng Lu, Chen Zhou, Jing Wu, Hao Jiang, and Songyue Cui. 2016. Integrating Granger Causality and Vector Auto-Regression for Traffic Prediction of Large-Scale WLANs. *KSII Transactions on Internet and Information Systems* 10, 1 (Jan. 2016), 136–151.

[28] Yingtao Luo, Qiang Liu, and Zhaocheng Liu. 2021. Stan: Spatio-temporal attention network for next location recommendation. In *Proceedings of the Web Conference 2021*. 2177–2185.

[29] Yisheng Lv, Yanjie Duan, Wenwen Kang, Zhengxi Li, and Fei-Yue Wang. 2015. Traffic Flow Prediction With Big Data: A Deep Learning Approach. *IEEE Transactions on Intelligent Transportation Systems* 16, 2 (April 2015), 865–873. https://doi.org/10.1109/TITS.2014.2345663

[30] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. *arXiv preprint arXiv:2211.14730* (2023).

[31] Yamila M. Omar and Peter Plapper. 2020. A Survey of Information Entropy Metrics for Complex Networks. *Entropy* 22, 12 (Dec. 2020), 1417. https://doi.org/10.3390/e22121417

[32] Cheonbok Park, Chunggi Lee, Hyojin Bahng, Yunwon Tae, Seungmin Jin, Kihwan Kim, Sungahn Ko, and Jaegul Choo. 2020. ST-GRAT: A Novel Spatio-Temporal Graph Attention Networks for Accurately Forecasting Dynamically Changing Road Speed. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20)*. Association for Computing Machinery, New York, NY, USA, 1215–1224. https://doi.org/10.1145/3340531.3411940

[33] Martin Rosvall and Carl T Bergstrom. 2008. Maps of random walks on complex networks reveal community structure. *Proceedings of the national academy of sciences* 105, 4 (2008), 1118–1123.

[34] Chao Shang, Jie Chen, and Jinbo Bi. 2022. Discrete Graph Structure Learning for Forecasting Multiple Time Series. In *International Conference on Learning Representations*.

[35] Claude Elwood Shannon. 1948. A mathematical theory of communication. *The Bell system technical journal* 27, 3 (1948), 379–423.

[36] Zezhi Shao, Zhao Zhang, Fei Wang, Wei Wei, and Yongjun Xu. 2022. Spatial-Temporal Identity: A Simple yet Effective Baseline for Multivariate Time Series Forecasting. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 4454–4458. https://doi.org/10.1145/3511808.3557702

[37] Ke Sun, Tieyun Qian, Tong Chen, Yile Liang, Quoc Viet Hung Nguyen, and Hongzhi Yin. 2020. Where to go next: Modeling long-and short-term user preferences for point-of-interest recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 214–221.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Advances in Neural Information Processing Systems*, Vol. 30. Curran Associates, Inc.

[39] Beibei Wang, Youfang Lin, Shengnan Guo, and Huaiyu Wan. 2021. GSNet: learning spatial-temporal correlations from geographical and semantic aspects for traffic accident risk forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4402–4409.

[40] Jingyuan Wang, Jiawei Jiang, Wenjun Jiang, Chao Li, and Wayne Xin Zhao. 2021. LibCity: An Open Library for Traffic Prediction. In *Proceedings of the 29th International Conference on Advances in Geographic Information Systems (SIGSPATIAL '21)*. Association for Computing Machinery, New York, NY, USA, 145–148. https://doi.org/10.1145/3474717.3483923

[41] Senzhang Wang, Jiannong Cao, and S Yu Philip. 2020. Deep learning for spatio-temporal data mining: A survey. *IEEE transactions on knowledge and data engineering* 34, 8 (2020), 3681–3700.

[42] Yue Wang, Mingsheng Liu, Yongjian Huang, Haifeng Zhou, Xianhui Wang, Senzhang Wang, and Haohua Du. 2022. Knowledge-based and data-driven underground pressure forecasting based on graph structure learning. *International Journal of Machine Learning and Cybernetics* (2022), 1–16.

[43] Yuandong Wang, Hongzhi Yin, Hongxu Chen, Tianyu Wo, Jie Xu, and Kai Zheng. 2019. Origin-destination matrix prediction via graph convolution: a new perspective of passenger demand modeling. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 1227–1235.

[44] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting.

Advances in Neural Information Processing Systems 34 (2021), 22419–22430.

[45] Junran Wu, Xueyuan Chen, Ke Xu, and Shangzhe Li. 2022. Structural entropy guided graph hierarchical pooling. In *Proceedings of the International Conference on Machine Learning*. PMLR, 24017–24030.

[46] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. 2020. Connecting the Dots: Multivariate Time Series Forecasting with Graph Neural Networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20)*. Association for Computing Machinery, New York, NY, USA, 753–763. https://doi.org/10.1145/3394486.3403118

[47] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Graph WaveNet for Deep Spatial-Temporal Graph Modeling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. Association for the Advancement of Artificial Intelligence (AAAI), 1907–1913. https://doi.org/10.24963/ijcai.2019/264

[48] Jiangnan Xia, Senzhang Wang, Xiang Wang, Min Xia, Kun Xie, and Jiannong Cao. 2022. Multi-view Bayesian spatio-temporal graph neural networks for reliable traffic flow prediction. *International Journal of Machine Learning and Cybernetics* (2022), 1–14.

[49] Mingxing Xu, Wenrui Dai, Chunmiao Liu, Xing Gao, Weiyao Lin, Guo-Jun Qi, and Hongkai Xiong. 2021. *Spatial-Temporal Transformer Networks for Traffic Flow Forecasting.*

[50] Zhenyu Yang, Ge Zhang, Jia Wu, Jian Yang, Quan Z Sheng, Hao Peng, Angsheng Li, Shan Xue, and Jianlin Su. 2023. Minimum entropy principle guided graph neural networks. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 114–122.

[51] Huaxiu Yao, Xianfeng Tang, Hua Wei, Guanjie Zheng, and Zhenhui Li. 2019. Revisiting Spatial-Temporal Similarity: A Deep Learning Framework for Traffic Prediction. *Proceedings of the AAAI Conference on Artificial Intelligence* 33, 01 (July 2019), 5668–5675. https://doi.org/10.1609/aaai.v33i01.33015668

[52] Junchen Ye, Leilei Sun, Bowen Du, Yanjie Fu, and Hui Xiong. 2021. Coupled layer-wise graph convolution for transportation demand prediction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4617–4625.

[53] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. 2021. Do transformers really perform badly for graph representation? *Advances in Neural Information Processing Systems* 34 (2021), 28877–28888.

[54] Bing Yu, Haoteng Yin, and Zhanxing Zhu. 2018. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, Stockholm, Sweden, 3634–3640. https://doi.org/10.24963/ijcai.2018/505

[55] Zhuoning Yuan, Xun Zhou, and Tianbao Yang. 2018. Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 984–992.

[56] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. 2022. *Are Transformers Effective for Time Series Forecasting?*

[57] Guangjie Zeng, Hao Peng, Angsheng Li, Zhiwei Liu, Chunyang Liu, Philip S Yu, and Lifang He. 2023. Unsupervised Skin Lesion Segmentation via Structural Entropy Minimization on Multi-Scale Superpixel Graphs. *arXiv preprint arXiv:2309.01899* (2023).

[58] Xianghua Zeng, Hao Peng, and Angsheng Li. 2023. Effective and Stable Role-Based Multi-Agent Collaboration by Structural Information Principles. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 37. 11772–11780.

[59] Xianghua Zeng, Hao Peng, Angsheng Li, Chunyang Liu, Lifang He, and Philip S Yu. 2023. Hierarchical State Abstraction based on Structural Information Principles. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence*.

[60] Qi Zhang, Jianlong Chang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. 2020. Spatio-Temporal Graph Structure Learning for Traffic Forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 01 (April 2020), 1177–1185. https://doi.org/10.1609/aaai.v34i01.5470

[61] Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. 2020. T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems* 21, 9 (Sept. 2020), 3848–3858. https://doi.org/10.1109/TITS.2019.2935152

[62] Chuanpan Zheng, Xiaoliang Fan, Cheng Wang, and Jianzhong Qi. 2020. GMAN: A Graph Multi-Attention Network for Traffic Prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2020-04-03), Vol. 34. 1234–1241. Issue 01. https://doi.org/10.1609/aaai.v34i01.5477

[63] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 11106–11115.

[64] Dongcheng Zou, Hao Peng, Xiang Huang, Renyu Yang, Jianxin Li, Jia Wu, Chunyang Liu, and Philip S Yu. 2023. SE-GSL: A General and Effective Graph Structure Learning Framework through Structural Entropy Optimization. In *Proceedings of the ACM Web Conference 2023*. 499–510.