ESSEN: Improving Evolution State Estimation for Temporal Networks using Von Neumann Entropy

Anonymous Author(s)
Affiliation
Address
email

Abstract

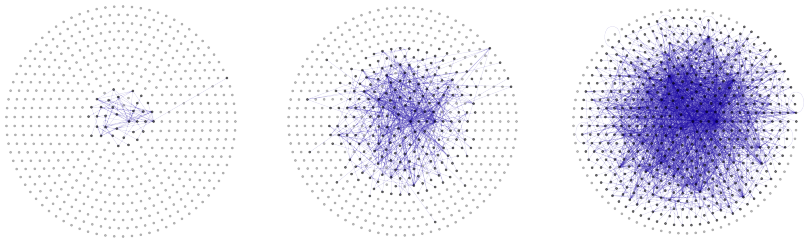
Temporal networks are widely used as abstract graph representations for real-world dynamic systems. Indeed, recognizing the network evolution states is crucial in understanding and analyzing temporal networks. For instance, social networks will generate the clustering and formation of tightly-knit groups or communities over time, relying on the triadic closure theory. However, the existing methods often struggle to account for the time-varying nature of these network structures, hindering their performance when applied to networks with complex evolving states. To mitigate this problem, we propose a novel framework called ESSEN, an **E**volution **S**tate**S** awar**E** **N**etwork, to measure temporal network evolution using von Neumann entropy and thermodynamic temperature difference. The developed framework utilizes the von Neumann entropy aware attention mechanism and network evolution state contrastive learning in the graph encoding. In addition, it employs a unique decoder Mixture of Thermodynamic Experts (MoTE) for decoding. ESSEN extracts local and global network evolution information using thermodynamic features and adaptively recognizes the network evolution states. Moreover, the proposed method is evaluated on link prediction tasks with transductive settings and inductive settings, with the corresponding results demonstrating its effectiveness compared to various state-of-the-art baselines [1].

1 Introduction

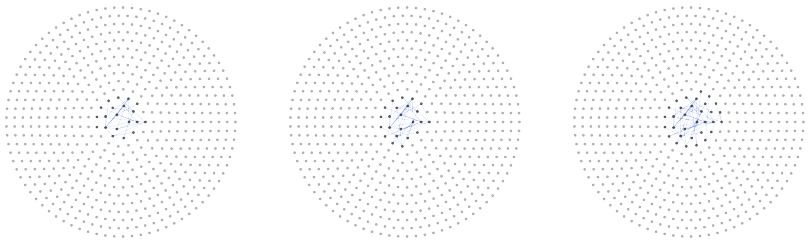
In recent years, graph representation learning has demonstrated excellent performance in a variety of static graphs [7; 14][28; 18]. Indeed, the success of static graph representation learning has led to a growing interest in continuous-time dynamic graph representation learning. Temporal network representation learning has emerged as an active research area focusing on learning low-dimensional representations that capture their topological and temporal properties. However, in many temporal networks, which are naturally generated in real-world systems, such as social networks[12] and citation networks, learning effective representations is still a difficult task. The evolving nature of these networks poses a significant challenge for network analysis and modeling, as the relationships between nodes and their properties evolve. The existing methods often struggle to account for the time-varying nature of these network structures, hindering their performance when applied to networks with complex evolving states. However, capturing the evolving states of temporal networks suffers from the following challenges: (1) Temporal networks have different types of evolving states, such as periodic, linear, or non-linear changes in their structure over time. Moreover, the evolving patterns can be changed in the network's different evolving stages. As illustrated in Fig.1, the evolution speed varies at different times in both datasets. The MathOverflow network evolves rapidly and has obvious central nodes, which means hot issues will receive long-term attention.

[1] Code is available at https://github.com/h1o2n3/ESSEN

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.



(a) MathOverflow: 10th Day (b) MathOverflow: 20th Day (c) MathOverflow: 30th Day



(d) BitcoinOTC: 10th Day (e) BitcoinOTC: 20th Day (f) BitcoinOTC: 30th Day

Figure 1: Network snapshots of the MathOverflow website and BitcoinOTC trading platform on the 10th, 20th, and 30th day. The black nodes represent the users who have connections, and the grey nodes represent the users with no edges prior to the snapshot time. The snapshots illustrate that temporal networks have different evolutionary states at different stages.

Instead, in the BitcoinOTC network, early active users may quickly become silent. The phenomenon requires algorithms that can perform extensive and effective recognition of diverse evolutionary states. (2) As time passes, temporal networks tend to accumulate more nodes and edges, resulting in an increasing number of possible connections and a rapidly growing neighborhood for each node. This growth in the neighborhood can lead to significant computational challenges when analyzing and modeling evolving patterns. Besides, many connections can quickly make structure recognization computationally infeasible in large and complex networks, especially the methods based on the anonymous walk in recent years[24; 6; 20]. The time complexity is tightly related to the length and number of paths, so it is challenging to balance time consumption and effectiveness.

To overcome these shortcomings, we aim to capture the evolution states using a thermodynamic entropy view. Thermodynamic network entropy is a macroscopic representation of network structures widely used to characterize the salient features of static and dynamic network systems in biology, physics, and social sciences. One of the most sophisticated studies involves the von Neumann entropy, which has been successfully used to describe the structural properties of random, small-world, and scale-free networks [1; 2]. Unfortunately, computing the required network entropies on the temporal network can be computationally burdensome due to spectral decomposition. Hence, we expand the approximate von Neumann entropy into the temporal network. Compared with other graph entropy, approximating von Neumann entropy with low time complexity can better adapt to the constantly evolving complex temporal networks. Moreover, we compute the important approximate thermodynamic quantity temperature for the temporal network. Measuring the thermodynamic temperature difference and the von Neumann entropy will provide a better understanding of the network's evolving state at any time.

On this basis, we propose a novel framework named ESSEN, an **E**volution **S**tate**S** awar**E** **N**etwork. ESSEN encodes node embeddings with evolution information by utilizing two proposed techniques: von Neumann entropy aware attention mechanism and virtual evolution node representation learning. In addition, ESSEN employs a unique decoder mixture of thermodynamic experts (MoTE) for decoding. Specifically, we project the global network's von Neumann entropy into each edge. The proposed von Neumann entropy aware attention mechanism aggregates the neighborhood in the virtual evolution graph and the original graph based on the von Neumann edge entropy. The virtual evolution graph is defined as supposing the node pair's test connection will truly happen at the specified test time future. The MoTE decoder evaluates the evolutionary state based on the thermodynamic temperature difference and von Neumann entropy in two graph views and provides a comprehensive result from

multiple experts. The decoder helps to recognize the network under various evolution states adaptively. Our framework is evaluated on transductive link prediction, inductive link prediction, and dynamic node classification tasks. The experimental results demonstrate our method's effectiveness compared to various state-of-the-art baselines. The overall contributions of our work are summarized as follows:

- To our best knowledge, we are the first to utilize the von Neumann entropy in temporal network representation learning. We provide a method to expand the approximate von Neumann Entropy and approximate thermodynamic temperature difference to temporal networks.
- We propose a novel framework, namely ESSEN. The model introduces a new perspective to encode evolution-aware node representations using the von Neumann entropy aware attention mechanism and virtual evolution node representation learning. Furthermore, the model uses a novel decoder MoTE that adaptively recognizes temporal network evolution states.
- We evaluate our framework on link prediction tasks with transductive and inductive settings. The results show the effectiveness of our proposed method compared to various state-of-the-art baselines.

2 Related Work

Temporal Network Representation Learning. Network representation learning is often used to transform large networks into lower-dimensional vectors. For instance, CTDNE [15] learns node embedding from a continuous-time dynamic network instead of a sequence of snapshots. Besides, JODIE [10] uses two recurrent neural networks (RNNs) to learn trajectories of users and items and updates the embedding when interaction occurs. TGAT [25] utilizes a self-attention mechanism and presents an encoding method to learn inductively. CAW [24] and NeuralTWs [6] learn temporal structure using random walk. Specifically, CAW proposes a new anonymization strategy, and NeuralTWs considers structural and tree traversal properties. TDLG[4] aims to model the edges in temporal networks directly instead of calculating from node embedding. However, it remains challenging to model global evolution under acceptable time complexity.

Von Neumann Entropy of Static Graph. The von Neumann entropy can be computed in the static graph using a quantum analogy[16]. According to this analogy, the Laplacian matrix [17] of a graph is interpreted as the density matrix [23] of an equivalent quantum system whose information content is given by the von Neumann entropy. The von Neumann entropy can be computed in the static graph as follows:

$$S_{VN} = -\sum_{i=1}^{|V|} \frac{\lambda_i}{|V|} \log \frac{\lambda_i}{|V|}, \tag{1}$$

where $\lambda_1, \ldots, \lambda_{|V|}$ are the eigenvalues of Laplacian matrix. This form of von Neumann entropy is effective for network characterization. The thermodynamics entropy could model the structure and complexity of a graph, where the von Neumann graph entropy[3; 16; 17] is often used to describe the statistical state of a network system. De Domenico et al.[5] use von Neumann graph entropy for the structural reduction in multiplex networks. Li et al.[11] study convergence through von Neumann entropy for network-ensemble comparison. Liu et al.[13] apply von Neumann graph entropy to study universal patterns of the dynamic genome. Besides, Wang[23] calculates the approximation of von Neumann graph entropy with node degrees to model the network evolution. However, applying von Neumann entropy to represent temporal structure is still scarce. To our best knowledge, there is almost no work using von Neumann entropy to characterize the temporal graph.

3 Preliminaries

Definition 1 Temporal Network. Formally, the temporal network can be denoted as $G = (V, E, T)$, where V represents the set of nodes, $E \subseteq V \times V$ represents the set of links, and T represents the set of timestamps. Each link (u, v, t) signifies a connection between node u and node v at time t. The temporal network evolves over time, with links appearing at different timestamps. The temporal networks can also include attributes associated with nodes or links, providing further information about the entities or their interactions at specific timestamps.

Definition 2 Dynamic Link Prediction. In a temporal network $G = (V, E, T)$, the dynamic link prediction task aims to predict the presence or absence of a link at a future timestamp based on the

observed network evolution history. Given a time window $T_w \subseteq T$, which contains the observed link data, denoted as $E_{T_w} \subseteq E$, the goal is to learn a function $f : (V, E_{T_w}, T_w) \to \{0, 1\}$ that assigns a probability score to the existence of a link (u, v) at the future timestamp t. Mathematically, the function f can be defined as:

$$f(u, v, t) = P(u, v|t, E_{T_w}), \tag{2}$$

where $P((u, v)|t, E_{T_w})$ represents the probability of the link (u, v) being present at the future timestamp t given the observed network and the link data E_{T_w} within the time window T_w.

Definition 3 Evolution States. Evolution states denote the specific arrangements of nodes and edges at specific time moments. These states can be characterized by network topology, reflecting the evolving nature of the network over time. For example, social networks at specific evolutionary states will generate the clustering and formation of tightly-knit groups or communities over time by triadic closure theory [29]. The theory is formally defined as $\exists u, v, w, w' \in V : (u, v), (v, w) \in E_{T_w}, (u, w), (u, w'), (v, w') \notin E_{T_w} \mapsto P(u, w|E_{T_w}) > P(u, w'|E_{T_w})$. Therefore, analyzing evolution states helps in understanding the temporal behavior of the network, identifying recurring patterns, predicting future states, and studying the impact of temporal dynamics on network properties and phenomena. In this paper, we aim to capture evolution states in a thermodynamic entropy view.

4 The Proposed Method

4.1 Evolution State Estimation

Von Neumann Entropy in Temporal Network. The von Neumann entropy can be computed in the static graph by Eq. 1. However, the application to temporal networks has two challenges: (1) The dynamic nature of the temporal network. Unlike static networks, where the structure remains constant, temporal networks capture evolving relationships and interactions. This dynamic nature introduces challenges in analyzing and modeling the network's behavior, as the network's topology and connectivity patterns may vary at different time points. (2) The expensive time complexity of computing the Laplacian eigenvalues. In Eq. 1, computing the eigenvalues of the Laplacian matrix is a computationally intensive task and the time complexity grows with the size of the network. The time cost is cubic in the number of nodes. In temporal networks, where the network structure changes over time, repeatedly calculating the Laplacian eigenvalues can become prohibitively expensive and time-consuming.

To solve these challenges, we must simplify the network and efficiently approximate thermodynamic quantities. **First**, we select a specific time interval from the temporal network and aggregate edge weights or frequencies, where the number of occurrences within the chosen time frame determines the strength of an edge. Following this process, the temporal network can be projected to the time-independent 2-D plane, which provides a simplified representation of the underlying network structure at a specific time. **Moreover**, inspired by [27], we use the approximate expression for the von Neumann entropy and reduce the computation to quadratic in the number of nodes. The approximate von Neumann entropy is

$$S_{VN}(G_t) = 1 - \frac{1}{|V|} - \frac{1}{|V|^2} \sum_{(u,v)\in E} \frac{1}{d_u d_v}, \tag{3}$$

where V is the node set of the temporal network, and d_u, d_v are the degree of node u and v at time t. This approximation allows the von Neumann entropy to be computed without explicitly solving the eigensystem for the normalized Laplacian. The proof of the approximate von Neumann entropy is presented in Appendix A. Thus, the von Neumann entropy can be computed in quadratic time using the node degrees for pairs of nodes connected by edges.

Thermodynamic Temperature Differences. In thermodynamics, a thermodynamic state of a system can be fully described by an appropriate set of principal parameters known as thermodynamic variables. Due to the dynamic nature of the temporal network, it is not sufficient to simply evaluate the evolution states for two moments using the von Neumann entropy. Thus, we introduce the thermodynamic temperature difference, defined as the rate of energy change with entropy between two networks, to complement the description of the network dynamics system. The expression which subject to the condition that the volume and number of particles are constant is shown as follows:

$$\mathcal{T}(G_1, G_2) = \frac{de}{dS} = \frac{e(G_1) - e(G_2)}{S(G_1) - S(G_2)}, \tag{4}$$

where $S(G_1)$ is the graph entropy on the graph G_1 and $e(G_1)$ is the graph average energy. The subject can be applied to temporal networks since the networks do not change significantly in node number during evolution. The thermodynamic temperature difference can bridge the graphs of two moments. More importantly, the thermodynamic temperature difference can also be approximated when we project the temporal network as a weighted graph, like in the computing process of von Neumann entropy. The approximate computation method uses low-order Taylor series that can be computed using the traces of powers of the normalized Laplacian matrix, avoiding explicit computation of the normalized Laplacian spectrum [26]. The detailed derivation process is in Appendix A. In summary, the temperature difference between the two networks can be approximated as

$$\mathcal{T}(G_1, G_2) = -\frac{2}{k} + \frac{2}{3k} \cdot \frac{\mathcal{K}(G_1) - \mathcal{K}(G_2)}{\mathcal{J}(G_1) - \mathcal{J}(G_2)}, \tag{5}$$

where

$$\mathcal{J}(G) = \sum_{u,v \in V} \frac{A_{uv}}{d_u d_v}, \tag{6}$$

$$\mathcal{K}(G) = \sum_{u,v,w \in V} \frac{A_{uv} A_{vw} A_{wu}}{d_u d_v d_w}, \tag{7}$$

k is the Boltzmann constant, and A is the adjacency matrix of the network. Especially we denote G'_{uv} as the graph G adds a new connection between u and v, the computation of $\mathcal{T}(G, G'_{uv})$ can be reduced as quadratic in the number of nodes. Because the expressions $\mathcal{J}(G'_{uv}) - \mathcal{J}(G)$ and $\mathcal{K}(G'_{uv}) - \mathcal{K}(G)$ can be rewritten as

$$\begin{aligned}
\mathcal{J}(G'_{uv}) - \mathcal{J}(G) &= \sum_{i \in V} \frac{A_{iv}}{d_i d_v (d_v + 1)} + \frac{A_{iu}}{d_i d_u (d_u + 1)}, \\
\mathcal{K}(G'_{uv}) - \mathcal{K}(G) &= \sum_{i,j \in V} \frac{A_{iv} A_{jv} A_{ij}}{d_i d_j d_v (d_v + 1)} + \frac{A_{iu} A_{ju} A_{ij}}{d_i d_j d_u (d_u + 1)} + \sum_{i \in V} \frac{A_{iv} A_{iu} A_{uv}}{d_i d_u d_v (d_u + 1)(d_v + 1)}.
\end{aligned} \tag{8}$$

In summary, by simplifying the network representation and using efficient approximations, the von Neumann entropy and thermodynamic temperature difference can be computed effectively in temporal networks. These measures provide insights into the evolving nature of the network and enable the estimation of its evolution state.

4.2 Evolution States Aware Graph Encoder

Von Neumann Entropy Aware Attention Mechanism. In graph encoding, we utilize the von Neumann entropy to explore a more diverse and balanced distribution of attention weights with the attention mechanism in the input neighborhood. This strategy helps the model learn network evolution states adaptively. According to Eq. 3, the global network entropy is a sum of contributions from individual edges. Thus, we decompose the global network entropy into components residing on the individual edges, so the von Neumann entropy of the edge connecting nodes u and v is

$$S_{VN}^{uv}(G_t) = \frac{1}{|E|} - \frac{1}{|V||E|} - \frac{1}{|E||V|^2} \frac{1}{d_u d_v}, \tag{9}$$

To better encode entropy features into attention layers, we incorporate the edge expression of von Neumann entropy via a bias term to the attention module[22]. Moreover, we use a time position encoding module[25] to supplement the continuous time information of edges simplified in the von Neumann entropy. Concretely, given a target node u at time t, the attention weight $\alpha_v^{(l)}$ from the neighbor node v on the l^{th} layer as

$$\alpha_v^{(l)} = \frac{Q_u^{(l)} \left(K_v^{(l)}\right)^T}{\sqrt{d}} + S_{VN}^{uv}(G_t), \tag{10}$$

$$Q_u^{(l)} = (h_u^{(l-1)} \| e_0 \| \phi(0)) W_Q, \tag{11}$$

$$K_v^{(l)} = M_{v,t}^{(l)} W_K, \tag{12}$$

$$M_{v,t}^{(l)} = \left({h_v}^{(l-1)} \| e_{uv,t} \| \phi(t_q - t) \right) W_M, \tag{13}$$

where d is the dimension of the node representation and "$\|$" is the concatenation operation. $W_K \in \mathbb{R}^{(d+d_t+d_e)\times d}$ and $W_Q \in \mathbb{R}^{(d+d_t+d_e)\times d}$ are the projection matrices to obtain the query matrices and key matrices. e_0 is an all-zero vector to keep the same dimension as K and V, and $\phi(*)$ is the generic time position encoding module from [25], which encodes the difference between the edge's timestamp and query timestamp. $M_{v,t}^{(l)}$ is the message representation at time t from node v to u, where $h_v^{(l-1)}$ is node v's hidden representation on the $(l-1)^{th}$ layer, $e_{uv,t} \in \mathbb{R}^{d_e}$ is the edge feature, and t_q is the query time.

Next, the model combines values with the attention weight aware of generating hidden representation $z_u^{(l)}(t)$ for node u. Finally, an MLP is used to combine the node representation of the previous layer with the neighborhood information:

$$h_u^{(l)} = MLP(h_u^{(l-1)} \| z_u^{(l)}(t)), \tag{14}$$

$$z_u^{(l)}(t) = \sum_{v \in \mathcal{N}_u} \mathrm{softmax}_v\left(\alpha_v(t)\right) V_v(t), \tag{15}$$

$$V_v^{(l)} = M_{uv,t}^{(l)} W_V, \tag{16}$$

where $V_v^{(l)}$ is the value vector of neighbor node v, and $\mathcal{N}_u$ is a neighbor node set that connects with node u before time t.

Virtual Evolution Node Representation Learning. Temporal networks follow evolution laws in the progress of time. The emergence of nodes and edges is often predictive, i.e., future network states can be predicted by past states and evolutionary laws. The network evolution state representation learning utilizes historical evolution information and the future virtual evolution graph to generate node representations. Specifically, the dynamic link prediction task aims to predict the probability of the link between two nodes appearing at a future moment. We suppose the link has been generated at the query moment and further construct the virtual evolution graph belonging to the query node pair on this suppose. For example, given a node pair (u, v) and the query time t, there is a virtual bridge at time t that connects the node u and v in the virtual evolution graph G'_{uv}. The approach makes the two node's neighborhoods interconnected. We denote h'_u and h'_v as the virtual future node embedding of u and v. Our framework will further measure the evolution differences through the decoding process.

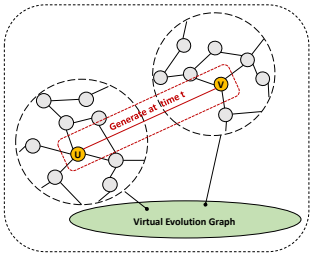


Figure 2: Virtual evolution graph of u, v at time t.

4.3 Mixture of Thermodynamic Experts Decoder

Expert Assessment Features. We use the von Neuman entropy of the original graph, the von Neuman entropy of the virtual evolution graph, and the thermodynamic temperature difference between both networks to combine a unique vector as expert assessment features. The vector represents the network's evolution states in the 3-D thermodynamic space, which is made up of current time and future time's von Neumann entropy and the thermodynamic temperature. We use Eq. 5 to approximate the thermodynamic temperature difference and Eq. 3 to compute the von Neumann graph entropy for the original and virtual evolution graphs. Furthermore, to control evolution-aware ranges, which is important for large temporal networks, we compute these approximate thermodynamic quantities in the node's neighborhood and set the sampled neighborhood size N.

Decoding Process. The mixture of thermodynamic experts decoder dynamically selects the appropriate thermodynamic expert model based on the input expert assessment feature vectors. For each expert, we use a two-layers MLP model to represent. Then the MoTE decoder combines the output embedding of each expert model using respective expert weights to produce the final target score as follows:

$$score(u, v, t) = \sum_{i=1}^{M} \sigma(W_i(h_u, h_v, h'_v - h_v, h'_u - h_u))\pi_i, \tag{17}$$

Table 1: Performance of AUC(%) for link prediction. The best results in each column are highlighted in bold font and the second-best results are underlined. We report the AP results in Appendix B.

Task	Methods	MathOverflow	BitcoinAlpha	BitcoinOTC	Wikipedia
Transductive	JODIE	86.07 ±0.48	91.14 ±0.18	92.29 ±0.11	94.62 ±0.50
	DyRep	80.77 ±0.65	79.39 ±3.17	79.21 ±4.10	94.59 ±0.20
	TGN	80.47 ±3.24	86.71 ±1.00	86.78 ±2.29	98.46 ±0.10
	TGAT	71.80 ±0.91	78.99 ±0.50	79.53 ±0.67	95.34 ±0.10
	CAW	53.82 ±0.28	64.70 ±0.93	73.95 ±1.22	98.96 ±0.10
	TDLG	84.02 ±0.16	92.83 ±0.22	93.48 ±0.22	88.93 ±0.09
	P-INT	54.09 ±3.24	88.11 ±0.60	91.26 ±0.69	98.78 ±0.10
	NeurTWs	92.56 ±0.51	93.95 ±0.41	95.75 ±0.01	94.54 ±0.87
	ESSEN	**98.60 ±0.40**	**99.10 ±0.16**	**98.88 ±0.42**	**99.03 ±0.33**
Inductive	JODIE	67.06 ±0.42	74.47 ±0.16	76.21 ±0.47	93.11 ±0.40
	DyRep	63.50 ±0.66	66.27 ±0.73	65.09 ±0.86	92.05 ±0.30
	TGN	64.50 ±1.17	69.36 ±0.94	76.52 ±1.25	97.81 ±0.10
	TGAT	60.02 ±0.75	66.42 ±1.17	66.62 ±1.99	93.99 ±0.30
	CAW	57.67 ±0.33	64.38 ±1.01	72.99 ±0.46	98.75 ±0.14
	TDLG	74.31 ±1.58	83.85 ±1.65	85.22 ±3.89	45.77 ±3.06
	P-INT	50.16 ±1.46	77.88 ±0.93	83.76 ±0.98	98.38 ±0.40
	NeurTWs	91.83 ±0.13	94.20 ±0.26	96.08 ±0.38	94.63 ±0.47
	ESSEN	**98.33 ±0.28**	**98.07 ±0.64**	**98.67 ±0.31**	**98.80 ±0.10**

$$\pi_i = softmax_i((\mathcal{T}(G, G'_{uv}) \| S_{VN}(G)) \| S_{VN}(G'_{uv}))W_\pi), \tag{18}$$

where M is the total number of experts, π_i is the mixing coefficient of expert i. $W_i \in \mathbb{R}^{4d\times 1}$ is the weight matrices for expert i. And $W_\pi \in \mathbb{R}^{3\times M}$ is the weight matrix of the gate unit. h_v and h'_v are the embeddings of node v in the original graph and virtual evolution graph generated by the encoder.

4.4 Optimization

During training, we evaluated the convergence behavior of our model by monitoring the training and validation loss, ensuring that the model was not underfitting or overfitting. The loss function is shown as follows:

$$\ell = \sum_{(v_i, v_j, t_{ij})\in\mathcal{E}} -\log P(v_i, v_j \mid t_{ij}) - \mathcal{Q}\cdot \mathbf{E}_{\tilde{v}\sim P(\tilde{v})} \log P(v_i, \tilde{v} \mid t_{ij}), \tag{19}$$

where (v_i, v_j, t_{ij}) is the observed edge on the temporal network, $\mathcal{Q}$ denotes the number of negative samples, and $P(\tilde{v})$ is the negative sampling distribution over the node space $\mathbf{E}$.

4.5 Computational Complexity Analysis

This section aims to highlight the efficiency of our approach in calculating approximate thermodynamic quantities of temporal networks. Based on Eq.3, Eq. 5, and Eq. 8, the time complexity of computing approximate von Neumann entropy and the approximate temperature difference is $O(|V|^2)$, where $|V|$ is the number of nodes in the network. Moreover, we compute the approximate thermodynamic quantities in the neighborhood for the large networks and set the sampled neighborhood size N. In this setting, the computational complexity can be reduced to $O(N^2)$, where N is the settable number. Therefore, the time complexity demonstrates scalability and feasibility for our method to operate in moderate or large networks. The controllable time complexity ensures efficient computation.

5 Experiments

5.1 Experimental Setup

Datasets. The temporal network datasets of our experiment are divided into three types: (a) QA: The "answers to questions" dataset of MathOverflow. (b) Bitcoin trading data: BitcoinAlpha Dataset and BitcoinOTC Dataset [9; 8]. (c) Social networks: Wikipedia Dataset[10]. Table 2 reports more details about these datasets.

Table 2: Statistics of the datasets.

Dataset	Nodes	Edges	Timespan
MathOverflow	21,688	107,581	2350 days
BitcoinOTC	5,881	35,592	1903 days
BitcoinAlpha	3,783	24,186	1901 days
Wikipedia	9,227	157,474	30 days

Baselines. In addition to reporting our ESSEN method's performance, we report results for several popular dynamic methods: a) JODIE [10]; b) DyRep [21]; c) TGAT [25]; d) TGN [19]; e) CAW [24]; f) TDLG[4]; g) P-INT[20]; and h) NeurTWs[6]. We report more details about baselines in Appendix C.

Link Prediction Task Settings. We evaluate our model on the link prediction task with two significant settings:

- **Transductive Setting.** The model under the transductive setting is trained on available nodes and their connections to predict links between these nodes in the future. The setting assumes the network will not add unseen nodes in the future test time. It mainly evaluates the model's transductive ability.
- **Inductive Setting.** The inductive setting predicts missing links for existing nodes and potential new nodes that may be added in the future. It generalizes link prediction beyond known nodes, considering the possibility of new nodes. It learns network patterns and characteristics to make predictions applicable to both known and unknown nodes.

Implementation Training Details. For each dataset, we used the training time points $T_{tr} = 70\%$ to split the dataset results in approximately 70%-15%-15% of the total edges [25]. The principal hyperparameters are set as follows: a) the number of attention heads $\mathcal{U} = \{2, 3\}$, b) the number of the GNN layers $\mathcal{L} = 2$, c) the maximum number of aggregated neighbors $n \in \{60, 80, 100\}$, d) the total number of experts in MoTE $M = \{4, 6, 8, 10\}$, and e) the dimension of the node embedding $D = 172$. We use the ADAM optimization algorithm for model training with a learning rate 1e-3 and batch size of 128. All the models are implemented in PyTorch and evaluated on a single Tesla A100 GPU.

5.2 Results and Discussion

Table 1 reports the transductive and inductive link prediction task results on four datasets, demonstrating our method's state-of-the-art performance on link prediction tasks. Indeed, our model significantly outperforms all baselines on all datasets. In particular, in the MathOverflow dataset, compared with NeurTWs, the second strongest baseline, ESSEN improved the AUC(%) by 5.04% and 6.50% on average on the transductive and inductive setting. The results demonstrate that our method has a clear advantage for temporal networks. Specifically, our method performs well on both long and short evolution time networks, while the effectiveness of baseline models varies significantly. CAW and TGAT have enormous performance gaps between MathOverflow and Wikipedia in all tasks. It indicates that our framework represents the network with ever-changing evolution states better. The superiority can be attributed to our von Neumann entropy aware mechanism, virtual evolution node representation learning, and MoTE decoder. In addition, our method is effective under both transductive and inductive settings. On the contrary, the baseline method JODIE cannot predict interactions well between unseen nodes because it pays more attention to node identities rather than the evolution states of temporal networks.

5.3 Ablation Study and Time Comparison

Ablation Study. To validate the effectiveness of the elements comprising ESSEN, we conduct a series of ablation studies and report the AUC results. We investigated the proposed modules with three ablated models on the Bitcoin-alpha dataset: a)ESSEN-E, the model removes the von Neumann edge entropy bias in the attention mechanism of ESSEN. b)ESSEN-V, the model removes virtual evolution node representation and

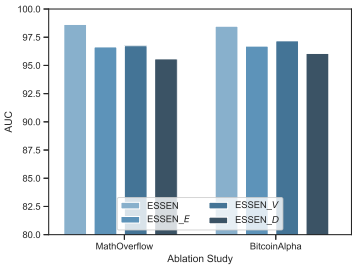
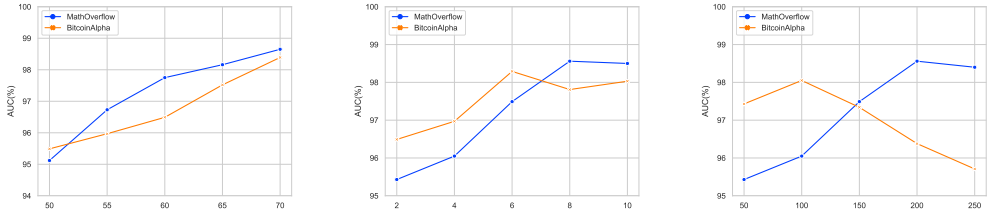


Figure 3: Ablation Study



(a) Training Time Point $T_{tr}(\%)$ (b) Number of Experts M (c) sampled Neighborhood Size N

Figure 5: Study on important settings. We report the results of the inductive link prediction.

only uses the node embeddings of the original graph for decoding. c)ESSEN-D, the model replaces the MoTE decoder with a simple MLP decoder. In Fig. 3, we can see the performance degradation without considering the von Neumann entropy information of edges, demonstrating the effectiveness of the proposed von Neumann entropy aware attention mechanism. Disabling the virtual evolution node representation also hurts performance. Furthermore, when the MoTE decoder is removed, BitcoinOTC and BitcoinAlpha exhibit more severe performance drops, demonstrating that the MoTE decoder excels on temporal networks with a long time span and more evolution states.

Time Comparison. Fig.4 compares the training times of ESSEN against the second-strongest baseline NTW. For fairness, we use the same batch size for both models and experiment in the same environment. Note that the running time of ESSEN is down quickly because the approximate thermodynamic quantities have been computed at the first epoch and use cache after that. If we pre-computed thermodynamic quantities for the model, ESSEN would run considerably faster than NTW.

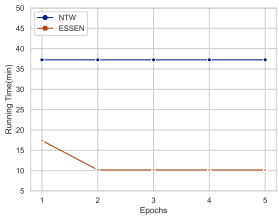


Figure 4: Time Comparison.

5.4 Parametric Sensitivity

we investigate the sensitivity of our ESSEN to various parameters and evaluate their impact on the model's performance. In Fig. 5, We report the results and have the following observations: a) Through the exploration of different training time points, our model keeps excellent performance, even if the number of samples in the training set is reduced since the training time shift. Especially, lowering the training time point means increasing the testing of more samples in the future. It clearly shows the robustness of ESSEN for complex evolutionary states. b) Regarding the number of experts M, there are sweet spots in both datasets. This finding indicates that different datasets exhibit a preference for specific numbers of experts in the MoTE decoder, which can be attributed to the varying complexity of evolutionary states in the temporal network. c) We observe a strong correlation between the evolution-aware size N and the node number of the temporal network. For instance, the MathOverflow with more number nodes has better performance with the large evolution-aware neighborhood. On the contrary, it negatively impacts the performance of BitcoinAlpha when N is higher than 100.

6 Conclusion

In this paper, we propose ESSEN, an **E**volution **S**tate**S** awar**E** **N**etwork for recognizing and analyzing the evolution states on temporal networks. We addressed the limitations of existing methods in capturing the time-varying nature of network structures, especially in complex evolving states. Our framework incorporates a von Neumann entropy aware attention mechanism and network evolution state contrastive learning in the graph encoding. The decoding stage utilizes a unique decoder called Mixture of Thermodynamic Experts (MoTE). We evaluated ESSEN on link prediction tasks in transductive and inductive settings and compared it to state-of-the-art baselines. The experimental results demonstrate the effectiveness of our proposed method in capturing temporal dynamics and outperforming existing approaches. Our work contributes to advancing the field of temporal network analysis and opens up possibilities for future research in other domains and additional network dynamics. In the future, we will focus on improving ESSEN's efficiency and scalability, allowing it to handle larger datasets and real-time analysis.

References

[1] Alstott, J., Pajevic, S., Bullmore, E., Plenz, D.: Opening bottlenecks on weighted networks by local adaptation to cascade failures. Journal of Complex Networks **3**(4), 552–565 (2015)

[2] Anand, K., Krioukov, D., Bianconi, G.: Entropy distribution and condensation in random networks with a given degree distribution. Physical Review E **89**(6), 062807 (2014)

[3] Braunstein, S.L., Ghosh, S., Severini, S.: The laplacian of a graph as a density matrix: a basic combinatorial approach to separability of mixed states. Annals of Combinatorics **10**(3), 291–317 (2006)

[4] Chanpuriya, S., Rossi, R.A., Kim, S., Yu, T., Hoffswell, J., Lipka, N., Guo, S., Musco, C.: Direct embedding of temporal network edges via time-decayed line graphs. In: ICLR (2022)

[5] De Domenico, M., Nicosia, V., Arenas, A., Latora, V.: Structural reducibility of multilayer networks. Nature communications **6**(1), 1–9 (2015)

[6] Jin, M., Li, Y.F., Pan, S.: Neural temporal walks: Motif-aware representation learning on continuous-time dynamic graphs. In: NeurIPS (2022)

[7] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR (2016)

[8] Kumar, S., Hooi, B., Makhija, D., Kumar, M., Faloutsos, C., Subrahmanian, V.: Rev2: Fraudulent user prediction in rating platforms. In: WSDM. pp. 333–341. ACM (2018)

[9] Kumar, S., Spezzano, F., Subrahmanian, V., Faloutsos, C.: Edge weight prediction in weighted signed networks. In: ICDM. pp. 221–230. IEEE (2016)

[10] Kumar, S., Zhang, X., Leskovec, J.: Predicting dynamic embedding trajectory in temporal interaction networks. In: SIGKDD. ACM (2019)

[11] Li, Z., Mucha, P.J., Taylor, D.: Network-ensemble comparisons with stochastic rewiring and von neumann entropy. SIAM journal on applied mathematics **78**(2), 897–920 (2018)

[12] Liben-Nowell, D., Kleinberg, J.: The link-prediction problem for social networks. Journal of the American society for information science and technology **58**(7), 1019–1031 (2007)

[13] Liu, S., Chen, P.Y., Hero, A., Rajapakse, I.: Dynamic network analysis of the 4d nucleome. bioRxiv p. 268318 (2018)

[14] Monti, F., Boscaini, D., Masci, J., Rodola, E., Svoboda, J., Bronstein, M.M.: Geometric deep learning on graphs and manifolds using mixture model cnns. In: CVPR. pp. 5115–5124 (2017)

[15] Nguyen, G.H., Lee, J.B., Rossi, R.A., Ahmed, N.K., Koh, E., Kim, S.: Continuous-time dynamic network embeddings. In: WWW. pp. 969–976 (2018)

[16] Passerini, F., Severini, S.: The von neumann entropy of networks. Available at SSRN 1382662 (2008)

[17] Passerini, F., Severini, S.: Quantifying complexity in networks: the von neumann entropy. International Journal of Agent Technologies and Systems (IJATS) **1**(4), 58–67 (2009)

[18] Ren, H., Leskovec, J.: Beta embeddings for multi-hop logical reasoning in knowledge graphs. In: NeurIPS. vol. 33, pp. 19716–19726 (2020)

[19] Rossi, E., Chamberlain, B., Frasca, F., Eynard, D., Monti, F., Bronstein, M.: Temporal graph networks for deep learning on dynamic graphs. In: ICLR (2020)

[20] Souza, A., Mesquita, D., Kaski, S., Garg, V.: Provably expressive temporal graph networks. Advances in Neural Information Processing Systems **35**, 32257–32269 (2022)

[21] Trivedi, R., Farajtabar, M., Biswal, P., Zha, H.: Dyrep: Learning representations over dynamic graphs. In: ICLR (2019)

[22] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)

[23] Wang, J.: Statistical Mechanics for Network Structure and Evolution. Ph.D. thesis, University of York (2018)

[24] Wang, Y., Chang, Y.Y., Liu, Y., Leskovec, J., Li, P.: Inductive representation learning in temporal networks via causal anonymous walks. In: ICLR (2021)

[25] Xu, D., Ruan, C., Korpeoglu, E., Kumar, S., Achan, K.: Inductive representation learning on temporal graphs. In: ICLR (2020)

[26] Ye, C., Comin, C.H., Peron, T.K.D., Silva, F.N., Rodrigues, F.A., Costa, L.d.F., Torsello, A., Hancock, E.R.: Thermodynamic characterization of networks using graph polynomials. Physical Review E **92**(3), 032810 (2015)

[27] Ye, C., Wilson, R.C., Comin, C.H., Costa, L.d.F., Hancock, E.R.: Approximate von neumann entropy for directed graphs. Physical Review E **89**(5), 052804 (2014)

[28] Zhang, M., Chen, Y.: Link prediction based on graph neural networks. Advances in neural information processing systems **31** (2018)

[29] Zhou, L., Yang, Y., Ren, X., Wu, F., Zhuang, Y.: Dynamic network embedding by modeling triadic closure process. In: Proceedings of the AAAI conference on artificial intelligence. vol. 32 (2018)