Adaptive Neural Network for Node Classification in Dynamic Networks

Dongkuan Xu[1], Wei Cheng[2], Dongsheng Luo[1], Yameng Gu[1], Xiao Liu[1], Jingchao Ni[2], Bo Zong[2], Haifeng Chen[2], Xiang Zhang[1]

[1]The Pennsylvania State University

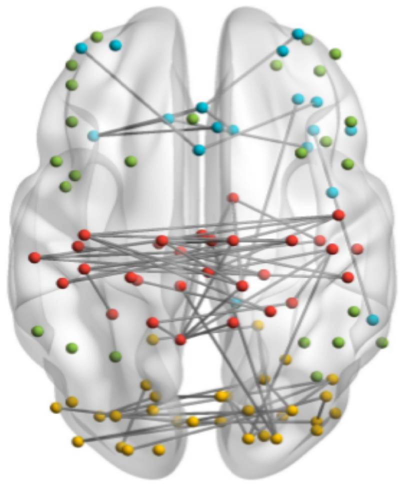
[2]NEC Laboratories America

PennState

NEC Laboratories America

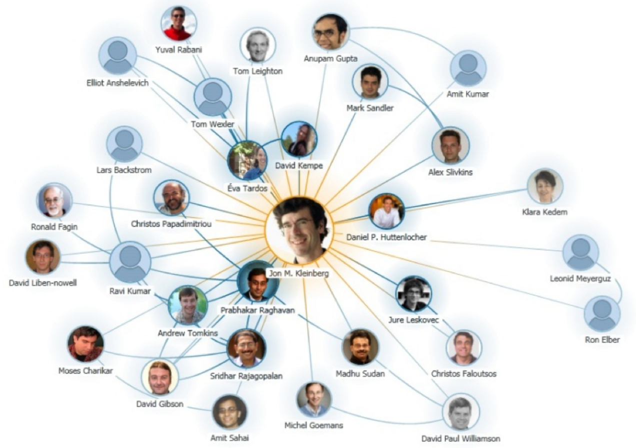
Relentless passion for innovation

Network-Structure Data is Prevalent

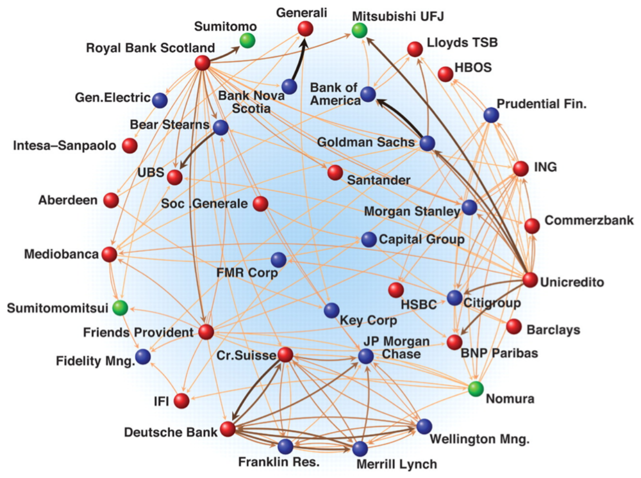


Brain Network

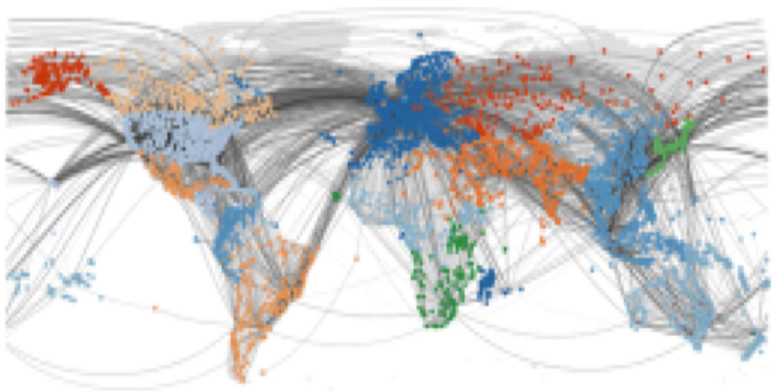
Social Network



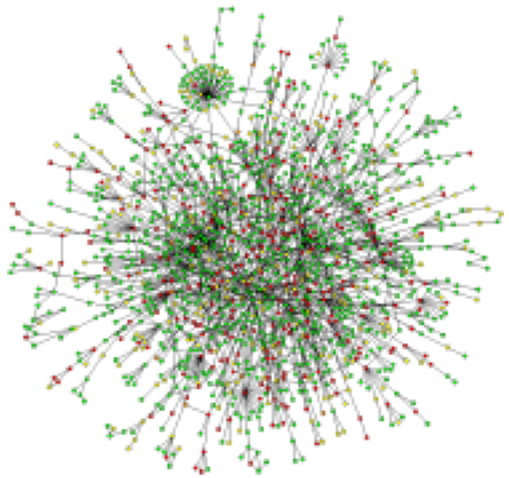
Co-author Network



Financial Network



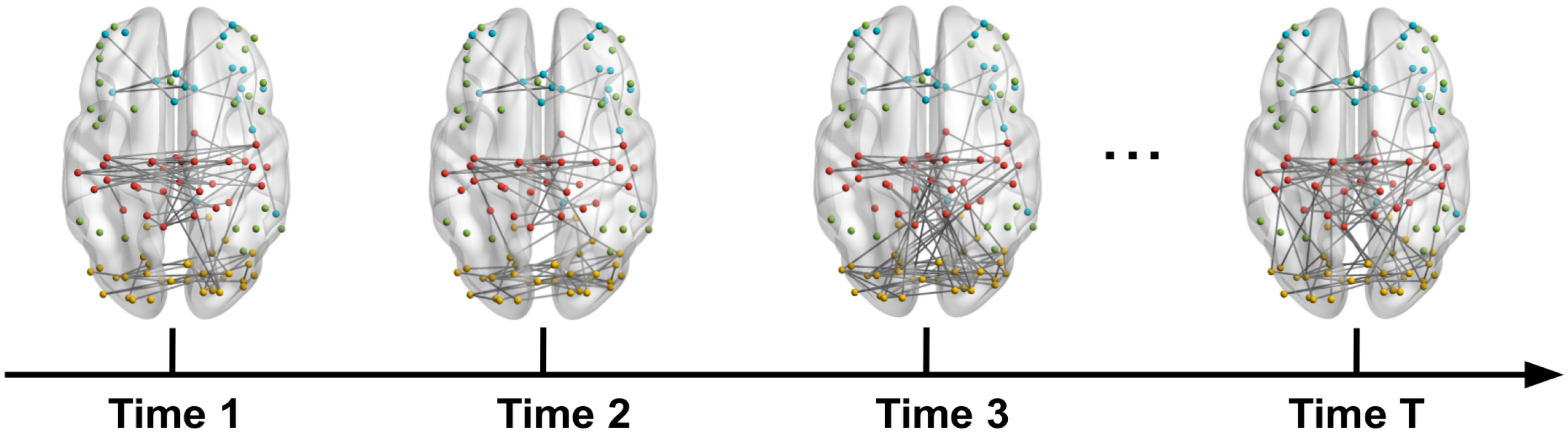
Traffic Network



Protein-Protein-Interaction Network

Dynamic Networks

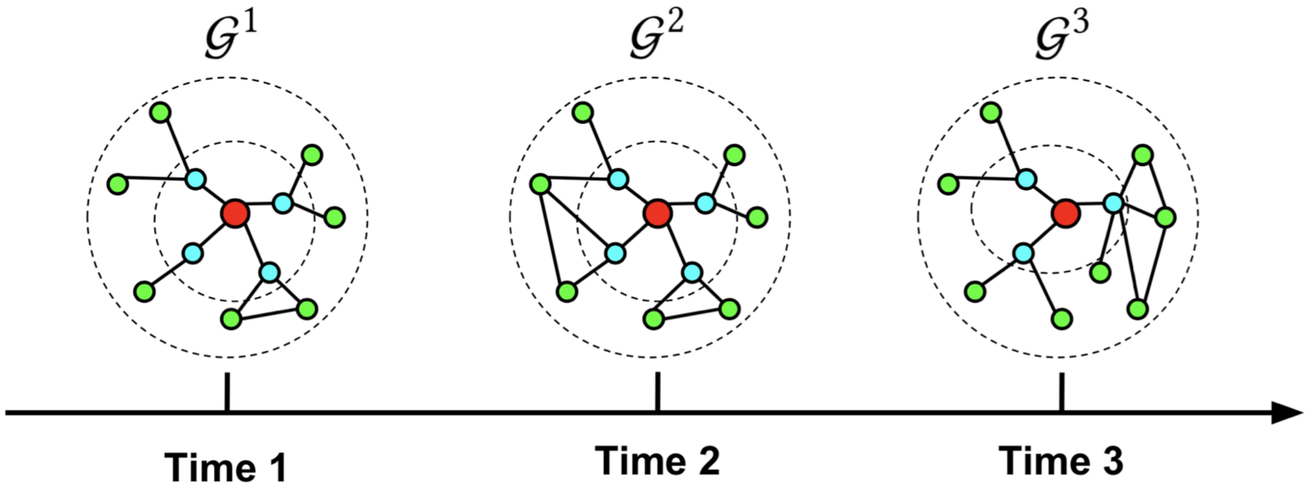
- Networks that vary over time
  - Network topology
  - Node attributes



Dynamic Brain Network

Problem Definition

- Node Classification in Dynamic Networks



Goal: What Is The Category of The Red Node

- Notations
  - Dynamic network $\mathbb{G} = (\mathcal{G}^1, \mathcal{G}^2, \cdots, \mathcal{G}^T)$, where $\mathcal{G}^t = (\mathcal{V}, \mathbf{A}^t, \mathbf{X}^t)$

- Definition
  - **Given:** $\mathbb{G}$ **and labels of a subset of nodes** $\mathcal{V}_L$
  - **Goal: to classify the nodes in subset** $\mathcal{V}_U$**, where** $\mathcal{V} = \mathcal{V}_L \cup \mathcal{V}_U$

Related Work

- **Static Networks**
  - Random walk based: DeepWalk [1]
  - Convolutional neural network based: GCN [2]
  - Local sampling based: GraphSAGE [3]
  - Neural attention mechanism based: GAT [4]

- **Dynamic Networks**
  - Matrix factorization based: DANE [5]
  - Triad structure based: DynamicTriad [6]
  - Point process based: HTNE [7]
  - Neural network based: STAR [8]

- **AdaNN**

  Dynamic networks

  Handle attributes & topology

  Differentiate different factors

1. Perozzi, B., Al-Rfou, R., & Skiena, S. (2014). Deepwalk: Online learning of social representations. In *SIGKDD*.
2. *Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.*
3. Hamilton, W., Ying, Z., & Leskovec, J. (2017). Inductive representation learning on large graphs. In *NeurIPS.*
4. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2018). Graph attention networks. In *ICLR*.
5. *Li, J., Dani, H., Hu, X., Tang, J., Chang, Y., & Liu, H. (2017). Attributed network embedding for learning in a dynamic environment. In CIKM.*
6. *Zhou, L., Yang, Y., Ren, X., Wu, F., & Zhuang, Y. (2018). Dynamic network embedding by modeling triadic closure process. In AAAI.*
7. *Zuo, Y., Liu, G., Lin, H., Guo, J., Hu, X., & Wu, J. (2018). Embedding temporal network via neighborhood formation. In SIGKDD.*
8. *Xu, D., Cheng, W., Luo, D., Liu, X., & Zhang, X. (2019). Spatio-temporal attentive RNN for node classification in temporal attributed graphs. In AAAI.*

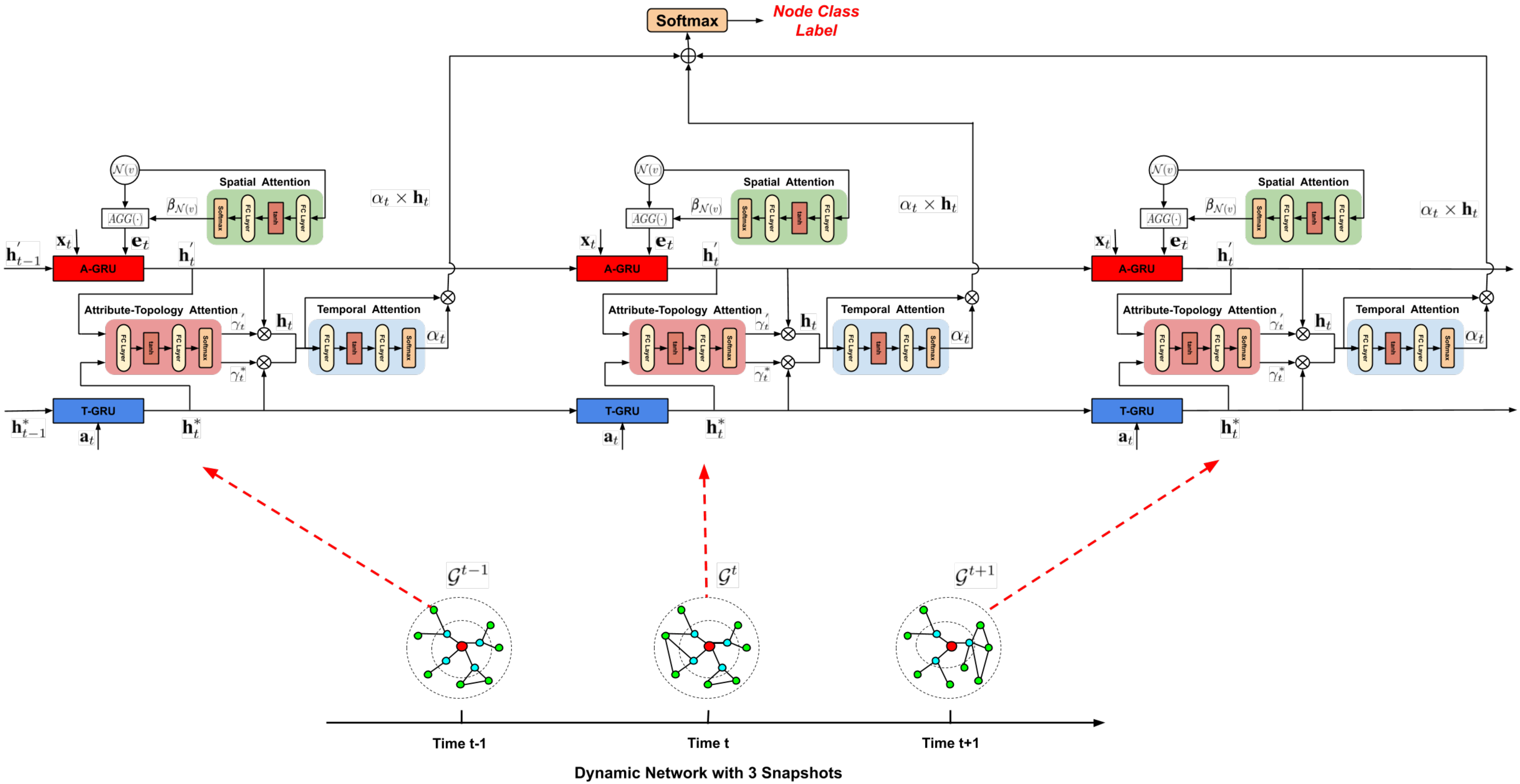
- **Challenges**
  - Spatial and temporal dimensions are entangled
  - Both node attributes and network topology evolve over time
  - Influence of attributes and topology varies for different networks

- **Motivations**
  - Consider attributes & topology at different time steps
  - Differentiate importance of different time steps
  - Differentiate importance of different neighbors
  - Differentiate importance of attributes and topology

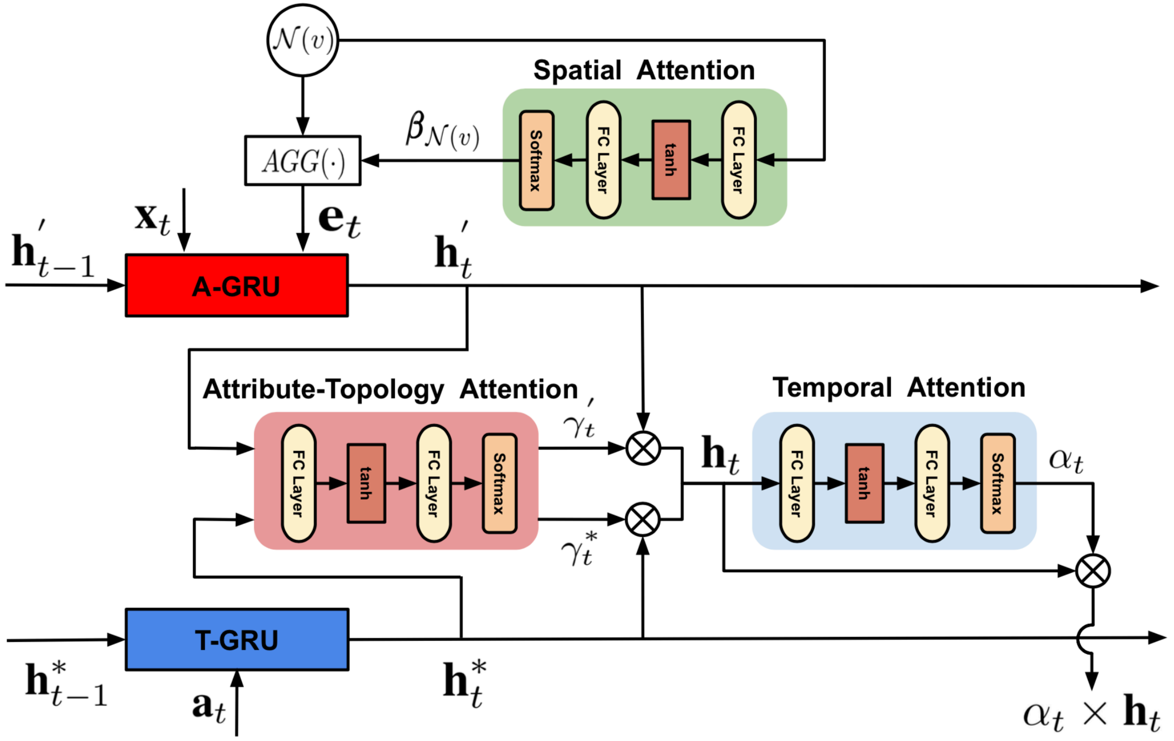
- **Challenges**
  - Spatial and temporal dimensions are entangled
  - Both node attributes and network topology evolve over time
  - Influence of attributes and topology varies for different networks
- **Motivations**
  - Consider attributes & topology at different time steps $\dashrightarrow$ **RNN Structure**
  - Differentiate importance of different time steps
  - Differentiate importance of different neighbors
  - Differentiate importance of attributes and topology

Triple Attention

Adaptive neural network for node classification in dynamic networks (AdaNN)

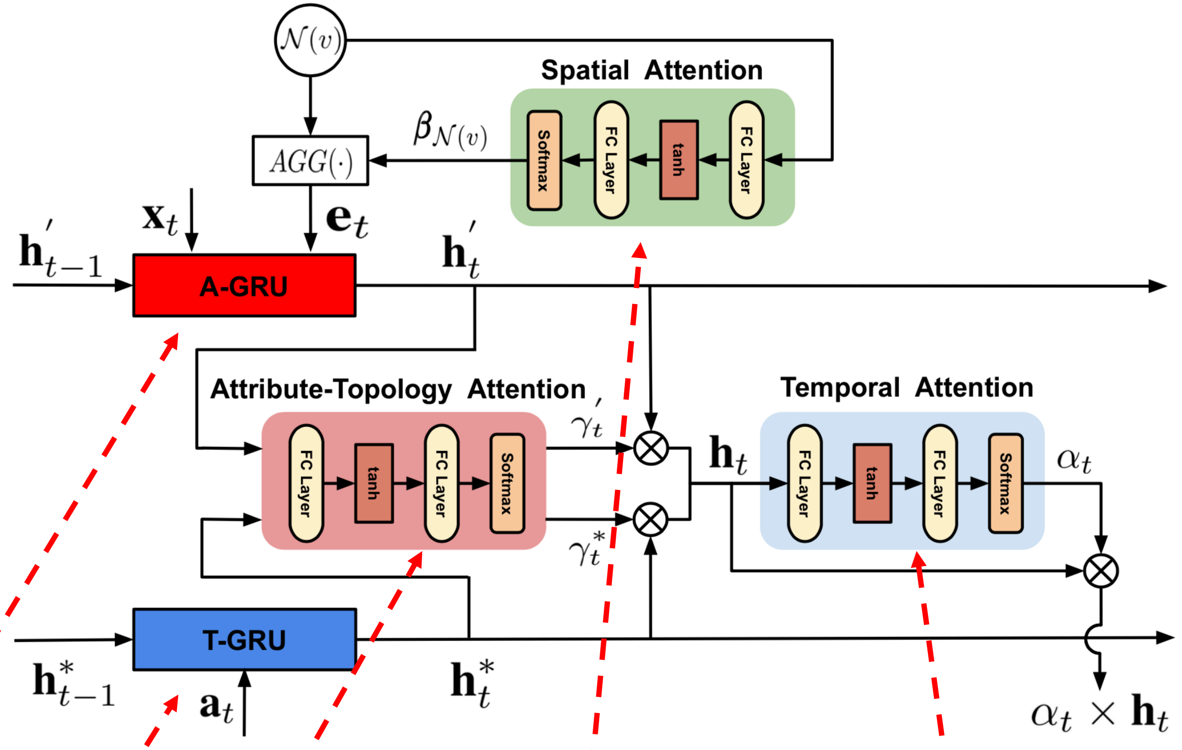


Architecture of AdaNN at each time step



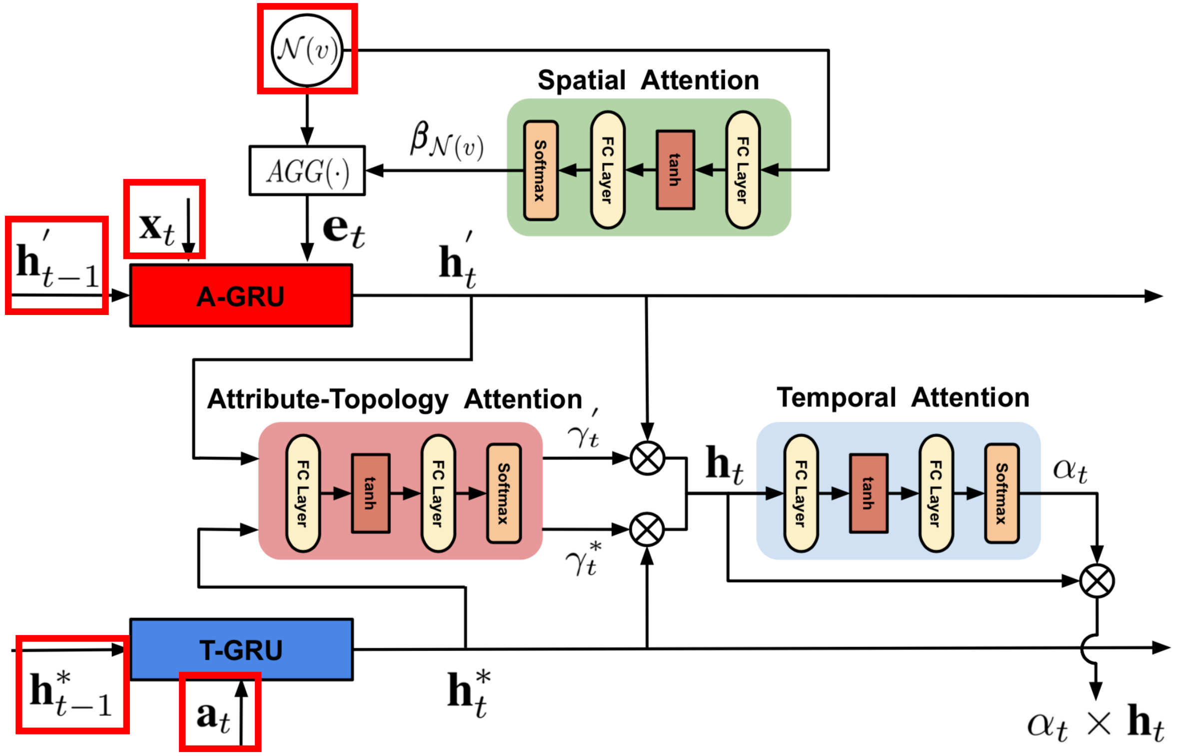
- Components:
  - A-GRU, T-GRU
  - Attribute-topology attention, Spatial attention, Temporal attention

Components of AdaNN



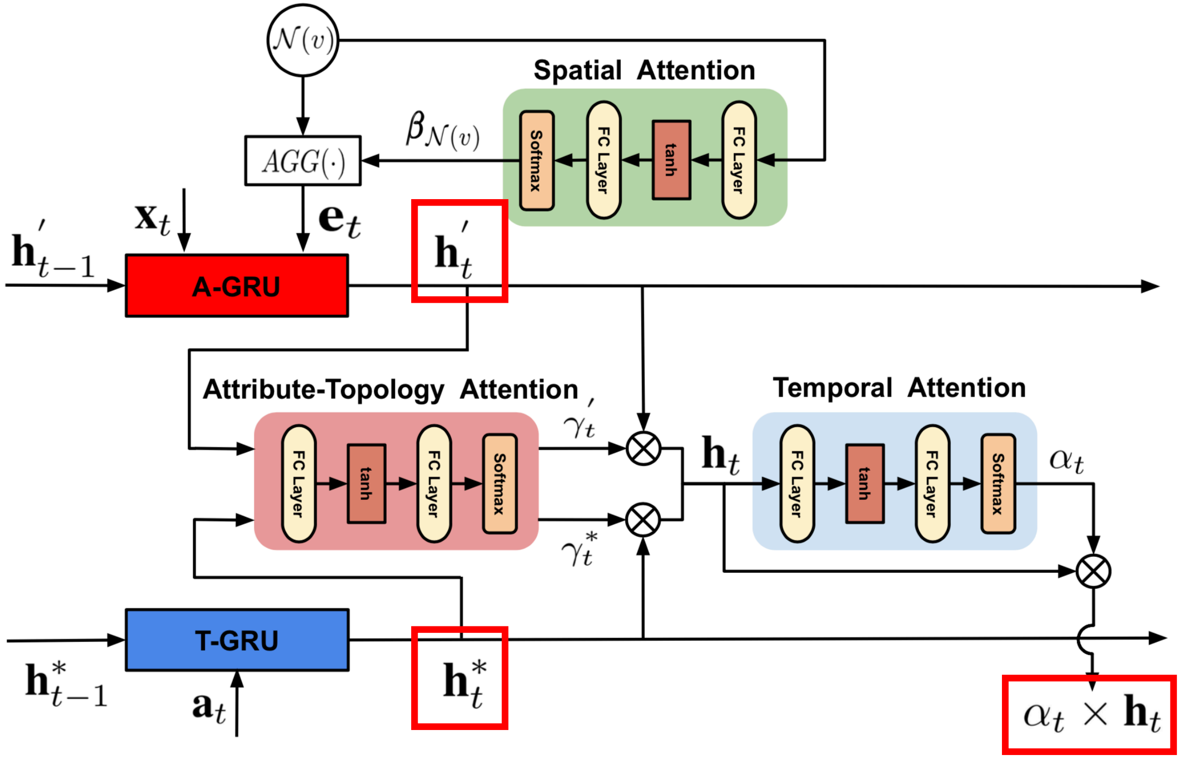
- Components:
  - A-GRU, T-GRU
  - Attribute-topology attention, Spatial attention, Temporal attention

Inputs at each time step



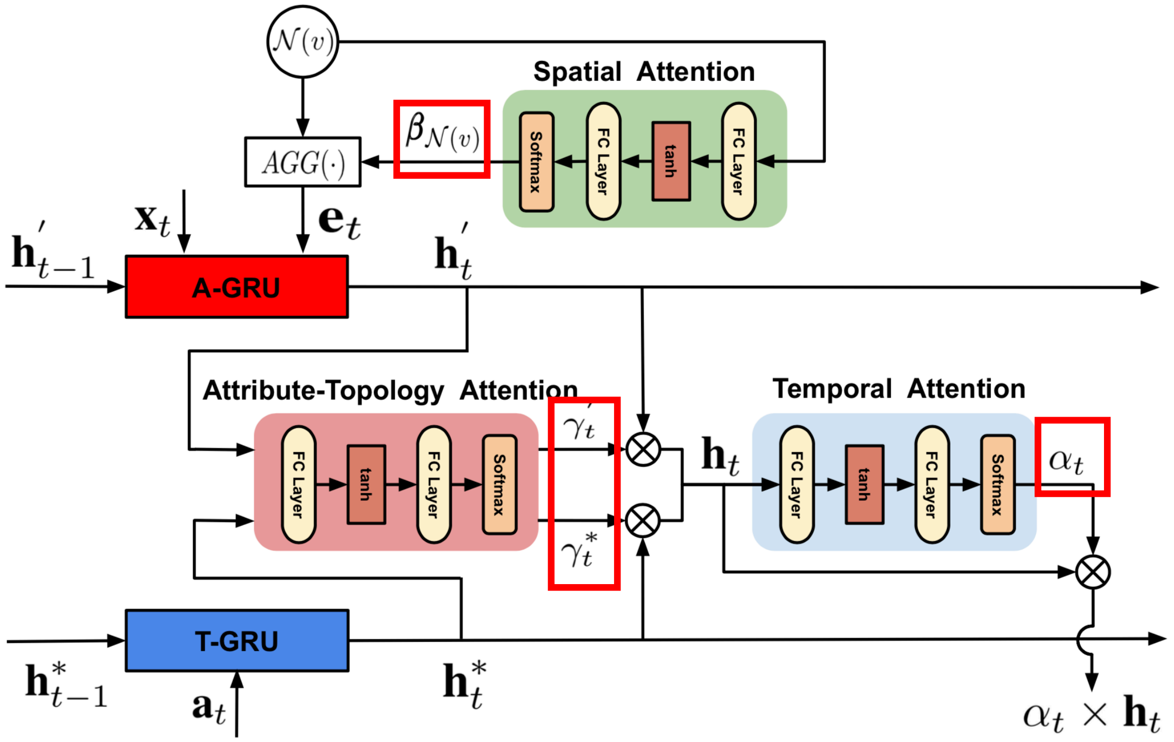
- Components:
  - A-GRU, T-GRU
  - Attribute-topology attention, Spatial attention, Temporal attention

Outputs at each time step



- Components:
  - A-GRU, T-GRU
  - Attribute-topology attention, Spatial attention, Temporal attention

Outputs of triple attention at each time step



- Components:
  - A-GRU, T-GRU
  - Attribute-topology attention, Spatial attention, Temporal attention

Calculation Detail of AdaNN

- Objective function

$$J = L_{ce} + \lambda P_{nn}$$

- T-GRU

$$\mathbf{z}_t^* = \sigma(\mathbf{W}_z^*[\mathbf{a}_t \oplus \mathbf{h}_{t-1}^*] + \mathbf{b}_z^*),$$
$$\mathbf{r}_t^* = \sigma(\mathbf{W}_r^*[\mathbf{a}_t \oplus \mathbf{h}_{t-1}^*] + \mathbf{b}_r^*),$$
$$\tilde{\mathbf{h}}_t^* = \tanh(\mathbf{W}_h^*[\mathbf{a}_t \oplus (\mathbf{r}_t^* \odot \mathbf{h}_{t-1}^*)] + \mathbf{b}_h^*),$$
$$\mathbf{h}_t^* = (\mathbf{1} - \mathbf{z}_t^*) \odot \mathbf{h}_{t-1}^* + \mathbf{z}_t^* \odot \tilde{\mathbf{h}}_t^*,$$

- A-GRU

$$\mathbf{z}_t^{'} = \sigma(\mathbf{W}_z^{'}[\mathbf{x}_t \oplus \mathbf{h}_{t-1}^{'} \oplus \mathbf{e}_t] + \mathbf{b}_z^{'}),$$
$$\mathbf{r}_t^{'} = \sigma(\mathbf{W}_r^{'}[\mathbf{x}_t \oplus \mathbf{h}_{t-1}^{'} \oplus \mathbf{e}_t] + \mathbf{b}_r^{'}),$$
$$\mathbf{s}_t^{'} = \sigma(\mathbf{W}_s^{'}[\mathbf{x}_t \oplus \mathbf{h}_{t-1}^{'} \oplus \mathbf{e}_t] + \mathbf{b}_s^{'}),$$
$$\tilde{\mathbf{h}}_t^{'} = \tanh(\mathbf{W}_h^{'}[\mathbf{x}_t \oplus (\mathbf{r}_t^{'} \odot \mathbf{h}_{t-1}^{'}) \oplus (\mathbf{s}_t^{'} \odot \mathbf{e}_t)] + \mathbf{b}_h^{'}),$$
$$\mathbf{h}_t^{'} = (\mathbf{1} - \mathbf{z}_t^{'}) \odot \mathbf{h}_{t-1}^{'} + \mathbf{z}_t^{'} \odot \tilde{\mathbf{h}}_t^{'},$$

- Spatial attention

$$\beta_u^k = \frac{\exp\{F(\mathbf{w}_k^\top[\mathbf{V}_k\mathbf{g}_{t(u)}^k \oplus \mathbf{V}_k\mathbf{g}_{t(v)}^k])\}}{\sum_{v' \in \mathcal{N}(v)} \exp\{F(\mathbf{w}_k^\top[\mathbf{V}_k\mathbf{g}_{t(v')}^k \oplus \mathbf{V}_k\mathbf{g}_{t(v)}^k])\}},$$

- Temporal attention

$$\alpha_t = \frac{\exp\{\tilde{\mathbf{w}}^\top \tanh(\tilde{\mathbf{V}}\mathbf{h}_t)\}}{\sum_{i=1}^T \exp\{\tilde{\mathbf{w}}^\top \tanh(\tilde{\mathbf{V}}\mathbf{h}_i)\}},$$

- Attribute-topology attention

$$\gamma_t^* = \frac{\exp\{\dot{\mathbf{w}}^\top \tanh(\dot{\mathbf{V}}\mathbf{h}_t^*)\}}{\exp\{\dot{\mathbf{w}}^\top \tanh(\dot{\mathbf{V}}\mathbf{h}_t^*)\} + \exp\{\dot{\mathbf{w}}^\top \tanh(\ddot{\mathbf{V}}\mathbf{h}_t^*)\}},$$
$$\gamma_t^{'} = \frac{\exp\{\dot{\mathbf{w}}^\top \tanh(\dot{\mathbf{V}}\mathbf{h}_t^{'})\}}{\exp\{\dot{\mathbf{w}}^\top \tanh(\dot{\mathbf{V}}\mathbf{h}_t^{'})\} + \exp\{\dot{\mathbf{w}}^\top \tanh(\dot{\mathbf{V}}\mathbf{h}_t^{'})\}},$$

Baseline Methods & Datasets

Comparison of baseline methods

	Method	Models Topology	Models Temporal	Models Neighborhood	Handles Attribute	Applies Attention
	DeepWalk	✓	×	✓	×	×
	GCN	✓	×	✓	✓	×
For static networks	GAT	×	×	✓	✓	✓
	GraphSAGE	✓	×	✓	✓	×
	node2vec	✓	×	✓	×	×
	LSTM	×	✓	×	✓	×
	GRU	×	✓	×	✓	×
For dynamic networks	DynGEM	✓	✓	×	×	×
	DynAERNN	✓	✓	✓	×	×
	DANE	✓	✓	✓	✓	×
	AdaNN	✓	✓	✓	✓	✓

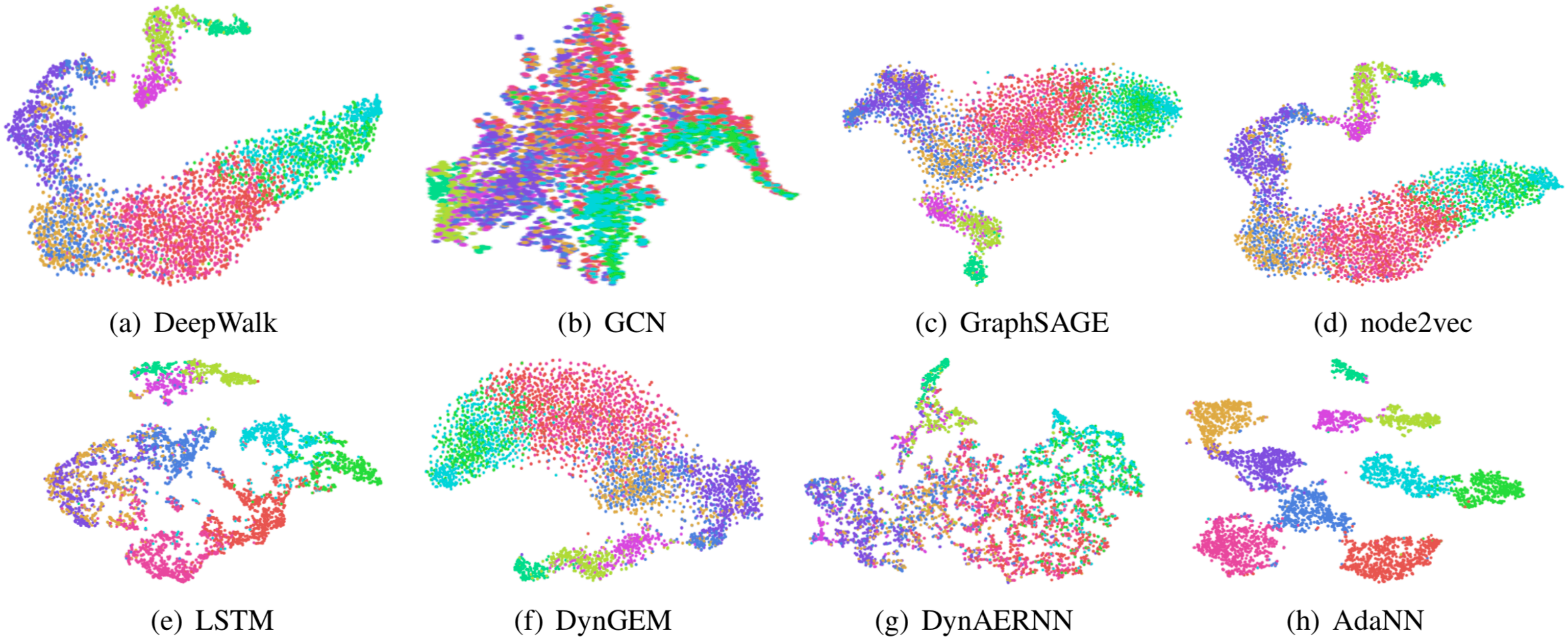
Description of the datasets

Dataset	# Nodes	# Edges	# Attributes	# Time Steps	# Categories
Brain	5000	1955488	20	12	10
DBLP-5	6606	42815	100	10	5
Epinions	16025	1144258	20	11	10
Reddit	8291	264050	20	10	4

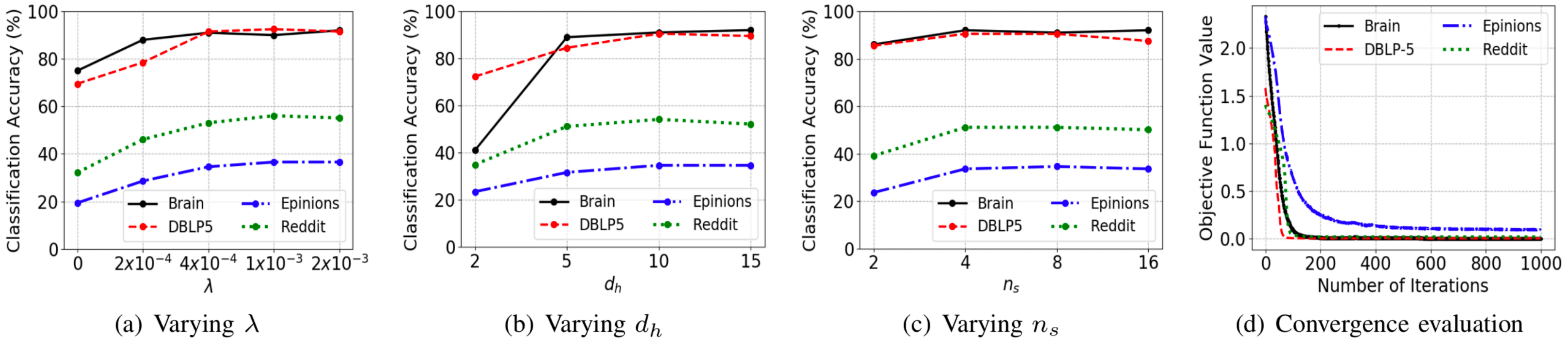
Experiment Results

Node classification comparison (%)

	Method	Brain			DBLP-5			Epinions			Reddit		
		ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1	ACC	AUC	F1
For static networks	DeepWalk	71.4	97.2	70.2	35.4	61.0	26.9	30.1	68.4	23.0	47.5	71.9	46.8
	GAT	43.8	86.2	49.2	32.5	48.6	26.1	22.5	63.1	18.3	29.6	52.4	21.8
	GCN	65.0	86.7	60.1	33.7	50.0	28.9	20.9	62.4	17.8	27.7	54.0	21.3
	GraphSAGE	69.4	96.7	74.1	71.0	90.7	69.7	24.5	63.9	18.7	42.5	66.8	42.5
	node2vec	71.0	96.8	70.6	36.9	64.2	27.2	32.8	70.2	26.0	48.0	72.2	47.9
For dynamic networks	GRU	80.4	98.2	80.2	75.6	91.5	75.2	17.3	61.7	17.2	42.1	67.2	41.9
	DynGEM	49.6	90.6	50.0	41.0	66.0	22.3	32.7	66.8	24.1	27.1	51.4	17.3
	DynAERNN	46.6	89.0	47.0	36.8	55.9	16.0	32.9	68.3	23.8	28.9	53.6	18.6
	DANE	85.2	94.8	85.9	82.5	92.3	81.6	31.8	67.1	23.5	45.7	70.0	45.1
	STAR	89.2	99.2	90.0	80.3	95.5	80.7	32.6	67.4	24.0	50.8	75.0	51.1
Variants	AdaNN-S	87.8	95.8	88.4	83.9	94.8	82.4	30.9	68.1	25.5	42.3	67.1	42.1
	AdaNN-P	85.1	91.5	84.7	81.5	93.7	81.7	27.5	62.8	22.4	46.1	71.3	46.2
	AdaNN-T	79.5	90.2	79.9	78.1	92.5	77.6	29.2	65.3	25.0	44.6	68.0	44.4
	AdaNN	**91.0**	**99.5**	**92.3**	**88.5**	**97.8**	**88.4**	**34.5**	**72.7**	**28.1**	**52.1**	**79.6**	**53.4**



Visualization results of the compared methods



Performance evaluation of AdaNN.

Thanks!

Q & A