



Spatial Temporal Graph Convolutional Network Model for Rumor Source Detection Under Multiple Observations in Social Networks

Xihao Wu¹, Hui Chen¹, Rong Jin², and Qiufen Ni¹(✉)

¹ School of Computer Science and Technology, Guangdong University of Technology, GuangZhou, China

niqiufen@gdut.edu.cn

² Department of Computer Science, California State University, Fullerton, USA

Abstract. Rumor source detection has long been an important but difficult problem. Most existing methods only rely on the limit observation of a single batch of single snapshot during the propagation process in the spatial graph networks, which neglects temporal dependency and temporal features of the rumor propagation process. Taking multiple batches of multiple snapshots as input can reveal the temporal dependency. Inspired by the traditional spatial-temporal graph convolution network (STGCN), which is a model can combine the spatial and temporal feature. In this paper, we propose an STGCN based model called Spatio-Temporal Approximate Personalized Propagation of Neural Predictions (STAPPNP), which firstly learns both the spatial and temporal features automatically from multiple batches of multiple snapshots to locate the rumor source. As there are no input algorithms which are suitable for multiple batches of multiple snapshots to capture the feature of nodes' connectivity in STAPPNP, we develop an input algorithm to generate a 4-dimensional input matrix from the multiple batches of multiple snapshots to feed the proposed model. Nonetheless, for deep learning models, such input of multiple batches of multiple snapshots results in a long training time as the number of model layers gets large. To address these issues, we improve the Spatio-Temporal-Convolutional(ST-Conv) block, in which we adopt the approximate personalized propagation of neural predictions in the spatial convolutional layer of STAPPNP. Our experimental results show that the accuracy of the source detection is improved by using STAPPNP, and the speed of the training process of STAPPNP outperforms state-of-the-art deep learning approaches under the popular epidemic susceptible-infected (SI) and susceptible-infected-recovery (SIR) model in social networks.

Keywords: Rumor source detection · Deep learning · Spatio-temporal graph convolutional networks · Social network

1 Introduction

Social networks have undergone rapid growth during the past a decade and have dramatically changed people’s lifestyle. In social networks, people share information easily, which has the advantage that people can communicate and learn through it at a low cost. But there are also disadvantages, the information can be false. Misinformation such as rumor can be spread rapidly and exponentially in social networks, which may lead to destructive consequences to individuals, organizations, and even society [1,2]. Consequently, the problem of rumor control via computational approaches has been studied by many researchers [3,4].

There are a variety of approaches to control the rumor propagation. In general, most research can be divided into two kinds of methods for this problem. One is content-based rumor identification and the other is rumor source detection. The major computational technique for content-based rumor identification [5] is the binary classification, which uses some machine learning model to classify whether a given text is a rumor or not. These models can be trained to classify the text by capturing the content features, sentiment features and user features from input texts. Detecting the rumor source in social networks is the second kind of method for controlling rumors. The rumor can be controlled if the rumor source is detected at an early stage of the rumor spread. The problem of rumor source detection has attracted lots of attentions in a decade ago [6]. In the past, most existing research on the rumor source detection problem only take a single snapshot observation of the rumor diffusion graph as input to estimate the source nodes, which neglects the temporal feature of rumor propagation. Due to the development of location technology, we can obtain multiple snapshots observed at different stages of the rumor propagation, and these independent snapshots can help reveal the temporal dynamics of the rumor propagation. Therefore, the research using multiple snapshots [4] has begun to appear recently. This fact motivates us to adopt a spatial-temporal graph convolutional network (STGCN) [7] framework that can combine temporal and spatial information at the same time in the multiple rumor propagation snapshots. However, the traditional STGCN model still suffers from shortcoming that the input of these models is either using original infection state as node label or one-dimensional one-hot encoded vector of the infection states, which limits observation in the underlying connectivity of nodes in the social network, and thus restricts the precision of source detection. Dong et al. [3] introduce an input algorithm instead of one-hot encoded to capture the feature of rumor centrality and source prominence based on connectivity of nodes in the social network, However, the input algorithm in reference [3] only for a single batch of single snapshot. For this problem, we introduce an input algorithm in our propose model Spatio-Temporal Approximate Personalized Propagation of Neural Predictions (STAPPNP), STAPPNP is a STGCN based model, in which the input algorithm is applied to multiple batches of multiple snapshots to generate a 4-dimensional vector for the input of node states in our proposed model. The 4-dimensional vector can capture the feature of rumor centrality and source prominence for each node. However, taking the multiple batches of multiple

snapshots as STAPPNP input requires a long training time. To address these issues, our propose model adopts the approximate personalized propagation of neural predictions (APPNP) [8] in the spatial layer that requires less training parameters than Graph convolutional network (GCN) based model, which enables much faster training speed. In this paper, we use the infection model [9] to describe our rumor propagation model in the social network. Additionally, we conduct extensive experiments on several real world social networks. The main contributions of this paper are summarized as follows:

- We firstly propose an input algorithm which is applicable to multiple batches of multiple snapshots input, which can capture the feature of rumor centrality and source prominence for each node.
- We use the approximate personalized propagation of neural predictions instead of graph convolutional network in spatial layers, which speeds up model training.

The rest of the paper is distributed as follows: Sect. 2 discusses the related work. Section 3 shows the structure and details of the proposed model, followed by the experiment and analysis in Sect. 4. Section 5 concludes the paper.

2 Related Work

2.1 Graph Convolutional Networks

GCN is a burgeoning deep learning model on graph. The earliest GCN is proposed by Duvenaud et al. [10], literature [10] introduce convolutional-like propagation rule on graphs. Defferrard et al. [11] use chebyshev polynomials to optimize the propagation rule, and the model achieved satisfactory experimental results. Kipf and Welling. [12] further approximate the propagation rule to a 1-order approximation, which reducing the computational complexity to linearity. However, 1-order approximation GCN still has the problem that training time becomes longer as the size of convolutional layers increases. Hence, Klicpera et al. [8] propose the Approximate Personalized Propagation of Neural Predictions (APPNP) algorithm to reduce the size of training parameters so as to improve the time of training.

2.2 Rumor Source Detection

Shah et al. [13] firstly study on the rumor source detection problem a decade ago. Wang et al. [14] investigate the problem of locating multiple source nodes, and propose the Label Propagation based Source Identification (LPSI) algorithm to identify them without knowledge of the underlying propagation model. Li et al. [15] then attempt to apply the deep learning model named Graph Convolutional Networks (GCN). However, Sha et al. [7] prove that the prediction precision of GCN based approach does not outperform the simple label propagation algorithm in literature [14] for a single snapshot. Hence, Sha et al. [7]

adopt a spatial temporal graph convolutional network (STGCN) to detect the source based on the multiple snapshots. But STGCN uses one-hot encoded as input algorithm which ignores the connectivity of nodes in social network. Dong et al. [3] introduce an input algorithm instead of one-hot encoded to capture the features of rumor centrality and source prominence based on connectivity of nodes in the social network.

3 Proposed Model

3.1 Input Algorithm for Multiple Batches of Multiple Snapshots

Inspired by literature [3], we propose our input algorithm to capture the feature of rumor centrality [14] and source prominence [16] from multiple batches of multiple snapshots. The process of our input algorithm is illustrated in Algorithm 1. Firstly, we concatenate the infection states of all nodes of multiple snapshots as the infection state matrix X . The shape of the matrix X is $Q \times T \times N$, and Q is the number of batches of multiple snapshots, T is the number of snapshots, N is the number of nodes on each snapshot. Each snapshot contains the infection states of all nodes in social network graph at different times. Infection state of random node $v = +1$ means the node is infected and $v = -1$ means the node is not infected for the SI model or SIR model (note that $v = -1 \in \{susceptible(S)state, recovery(R)state\}$, $v = +1 \in \{infected(I)state\}$). Next, from step 2 to step 16, we need to generate some matrix that can capture the feature of rumor centrality and source prominence. We reduce the dimensionality of X to a new 2-D matrix $X_1 \in R^{(Q \times T) \times N}$, where $R^{(Q \times T) \times N}$ denotes the matrix X_1 shape as $(Q \times T) \times N$, $(Q \times T)$ denotes the result of $Q \times T$. P and O are defined and initialized, they are used to generate c and d . The dimensions of a, b, c and d are same as X_1 , literature [3] proves that b can capture the source prominence, c and d can capture the features of rumor centrality. In step 14 to 16, the $\alpha \in (0, 1)$ is a hyperparameter, α denotes the percentage of label information aggregated from neighboring nodes. At step 17, a, b, c and d are concatenated into a high dimensional vector corresponding to each snapshot. Finally, in step 18, we obtain a 4-D output matrix Y^0 by reshaping the 3-D matrix Y' , where Y^0 denotes the infection state matrix for all nodes under the observation of multiple snapshots in $(Q \times T)$. The infection label of each node is a 4-dimensional vector, which contains the feature of rumor centrality and source prominence.

3.2 Spatio-Temporal-Convolutional Block

Inspired by [8], we propose an improved Spatio-Temporal-Convolutional (ST-Conv) block, which adopts the Approximate Personalized Propagation of Neural Predictions (APPNP) as the spatial convolutional layer to extract the spatial features of the nodes. We use APPNP instead of GCN as spatial layer can speed up the model training. As illustrated in Fig. 1, we take the output Y^0 of the Algorithm 1 as the first ST-Conv block input, and Y^0 is in the shape of

Algorithm 1. Input algorithm for multiple batches of multiple snapshots.

Require: Infection State Matrix $X \in R^{Q \times T \times N}$; degree matrix D ; adjacency matrix A ; hyperparameter α ;

Ensure: Four-Dimension infection state input matrix $Y \in R^{Q \times T \times N \times 4}$;

- 1: Defined the normalized matrix $U = D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$, Q is the batch size, T is the number of snapshots, N is the number of nodes, Identity matrix I ;
- 2: $X_1 = \text{reshape}(X) \in R^{(Q \times T) \times N}$
- 3: define $P = X_1, O = X_1$;
- 4: **for** $i = 0 \rightarrow \text{len}(Q \times T)$ **do**
- 5: **for** $j = 0 \rightarrow \text{len}(N)$ **do**
- 6: **if** $X_{1_{i,j}} == -1$ **then**
- 7: $P_{i,j} = 0$;
- 8: **else**
- 9: $O_{i,j} = 0$;
- 10: **end if**
- 11: **end for**
- 12: **end for**
- 13: $a = X_1$;
- 14: $b = (1 - \alpha)(I - \alpha U)^{-1}X_1$;
- 15: $c = (1 - \alpha)(I - \alpha U)^{-1}P$;
- 16: $d = (1 - \alpha)(I - \alpha U)^{-1}O$;
- 17: $Y' = \text{concatenate}(a, b, c, d) \in R^{(Q \times T) \times N \times 4}$;
- 18: $Y^0 = \text{reshape}(Y') \in R^{Q \times T \times N \times 4}$;
- 19: **return** Y^0 ;

$R^{Q \times T \times N \times C_{in}}$, where C_{in} denotes the number of input channels. As Algorithm 1 expands the label vector of each node to 4 dimensions, thus, the value of C_{in} in the first ST-Conv block is equal to 4. The details of our ST-Conv block are described as follows:

Temporal Gated-Convolution. The temporal convolutional layer contain a 1-D convolutional neural network (CNN) with a width- K_t kernel followed by gated linear unit (GLU) as activation function. Since the convolutional kernel size is K_t , the time series will be reduced by $K_t - 1$ after passing through a temporal convolutional layer. The temporal convolutional layer can be represented as follows:

$$Y_1 = \text{GLU}(\text{CNN}(Y^0)) \in R^{Q \times (T - K_t + 1) \times N \times C_h} \quad (1)$$

we use Eq. (1) to capture the temporal feature, in which Y^0 is the input of the temporal layer, C_h is the output dimension of the temporal layer, and Y_1 is the output matrix of the temporal layer, GLU is a gated linear unit and CNN is a convolutional neural network layer.

Spatial Approximate Personalized Propagation of Neural Predictions.

In the spatial layer we use APPNP instead of GCN to capture spatial features. The rules of spatial feature extraction are designed as follows:

$$H^{(0)} = Z = f_{\theta}(Y_1) \quad (2)$$

$$H^{(l+1)} = (1 - \delta)\tilde{A}H^{(l)} + \delta Z \in R^{Q \times (T - K_t + 1) \times N \times C_h} \quad (3)$$

$$Y^2 = H^{(L)} = \text{softmax}((1 - \delta)\tilde{A}H^{(L-1)} + \delta Z) \in R^{Q \times (T - K_t + 1) \times N \times C_h}. \quad (4)$$

The Eq. (2) using neural networks to predict the class of nodes, f_{θ} is a neural network with a parameter set θ . Y_1 represents the input of the spatial layer and corresponds to the output of the first temporal layer. The Eq. (3) is the message propagation rule, we use this equation to aggregate information about neighboring nodes. $H^{(l+1)}$ is the matrix of activations in the $(l+1)^{th}$. $\delta \in (0, 1]$ is a hyperparameter, which represents the restart probability (i.e., the percentage of label information aggregated from root nodes). In Eq. (4), L denotes the number of power iteration steps, and $l \in [0, L - 2]$ in Eq. (3). The APPNP uses *softmax* as the activation function, with the spatial layer kernel size K_s . Through APPNP, the spatial information of each node can be extracted and then integrated with its neighbors within the area of K_s layers. As shown in Fig. 1. After passing the spatial layer, we set the output Y_2 of the spatial layer as the input to the following temporal layer, the following temporal layer is the same as the first temporal layer in the ST-Conv block. Due to an ST-Conv block contains two gated sequential convolution layers and one spatial convolution layer, the time series will be subtracted by $(2K_t - 2)$ after passing through an ST-Conv block. Assume that we have L_1 ST-Conv blocks for feature extractions, then the time series of the final output matrix will be subtracted by $(2L_1K_t - 2L_1)$. Thus the final output dimension is $Y_3 \in R^{Q \times (T - 2L_1K_t + 2L_1) \times N \times C_{out}}$, where C_{out} denotes the output dimension of the last gated sequential convolution layer.

3.3 Spatio-Temporal Approximate Personalized Propagation of Neural Predictions

Our STAPPNP is built on the STGCN construction. The construction of STAPPNP is illustrated in Fig. 1. At first, the input is an infection state matrix X of the shape $Q \times T \times N$, which is generated based on multiple batches of multiple snapshots. The multiple batches of multiple snapshots are observations of the rumor propagation processes on social network graph. Next, we take X as the input of Algorithm 1, and the output is a new matrix Y^0 of the shape $Q \times T \times N \times 4$. Then, the matrix Y^0 flows via two ST-Conv blocks where each block contains a spatial layer and two temporal gated-convolutional layers to capture the spatial and temporal features of each node. Note that the number of ST-Conv blocks is not necessarily fixed. In practice, multiple ST-Conv blocks can be stacked together to extract the features of nodes. Finally, the output of the last ST-Conv block Y_3 is sent to the output layer. The temporal convolution layer attached with one fully connected layer to output the probability of each node contains the rumor source node. The activation of the fully connected layer is *softmax*. The output layer can be written as:

$$Y = \text{Softmax}(FC(GLU(CNN(Y_3)))) \in R^{Q \times N} \quad (5)$$

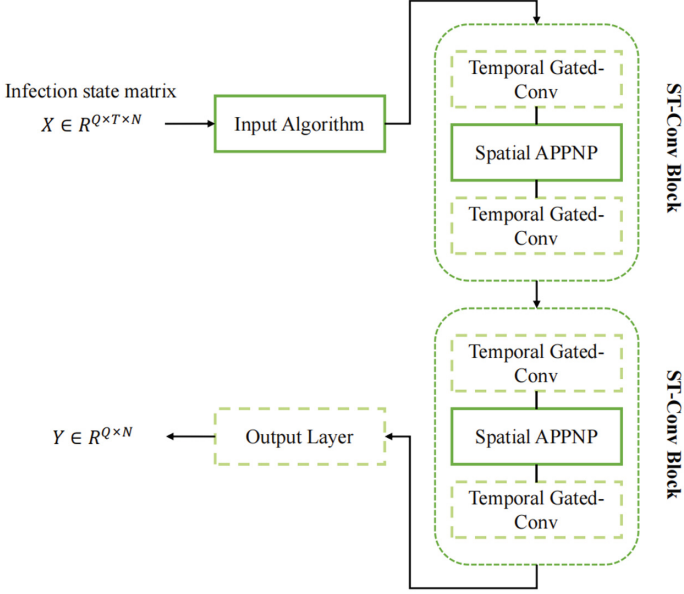


Fig. 1. Construct of STAPPNP.

the output of STAPPNP is a matrix Y , the shape of Y is $Q \times N$, which represents the probabilities that each node in Q batch graphs is the rumor source, and FC is a fully connected layer.

4 Experiment

4.1 Datasets

Our model is suitable for small or medium social networks. Datasets are divided into two types. The first type are ER and BA social networks [17]. The second type is empirical contact social network adopted from High school [18] and SFHH conference [19], both of which are generated on the dynamic contact sequences. The statistics of the datasets are listed in Table 1.

Table 1. Statistics of four datasets.

Datasets	Nodes	Edges
BA	2000	19900
ER	1000	9953
High School	774	7792
SFHH	403	9565

4.2 Parameter Setting

Common Settings. In our experiment, we generate 2000 rumor propagation processes for each social network based on SI and SIR infection propagation models. 80%, 10% and 10% of rumor propagation processes are used as the training dataset, the validation set, and the test set, respectively. The batch size is set to be 50. All the deep learning models are trained for 50 epochs using RMSProp optimizer, and the learning rate starts from 0.001 and decays with rate 0.7 every 5 epochs. Since the RSD problem is a multi-classification problem, cross-entropy is used as the loss function. As shown in Fig. 1, we use two ST-Conv blocks to capture features of nodes. The input channels and output channels for the first temporal layer and spatial layer and last temporal layer in first ST-Conv block are $\{4, 64\}$, $\{64, 64\}$, $\{64, 128\}$. The input channels and output channels for the second ST-Conv block are $\{128, 64\}$, $\{64, 64\}$, $\{64, 72\}$.

Parameter Setting Under Two Infection Propagation Models. Under the SI model, the infection rate is set to be 0.05, each rumor propagation process simulates 15 iterations of rumor propagations to generate 15 snapshots. Then, we randomly choose 10 rumor propagation infection snapshots. For a small batch of snapshots under the SI model, we set temporal kernel size (K_t) from set $\{2,3\}$ and spatial kernel size (K_s) from set $\{2,3,4,5\}$, we find that STAPPNP achieves the best performance with $K_t = 2$ and $K_s = 5$. Under the SIR model, we set the infection rate and the recovery rate to be 0.25 and 0.1, respectively. We simulate the rumor propagation process with 30 iterations to generate the 30 rumor propagation infection snapshots, and then we randomly select 16 snapshots as the inputs as well as we set temporal kernel size from set $\{2,3,4\}$ and spatial kernel size from set $\{2,3,4,5\}$. For the SIR model, the STAPPNP achieves the best performance with $K_t = 3$ and $K_s = 5$. In the following experiments, we adhere to this setup.

4.3 Comparison and Evaluation

We compare our STAPPNP with three popular baseline models in terms of two types of evaluation metrics, as shown in Table 2. All models are trained on NVIDIA GeForce RTX 2080 Ti GPU servers.

Evaluation Metrics. We evaluate the performance of the model with the following metrics:

- **Detection accuracy.** Detection accuracy is the percentage of true rumor sources being detected correctly. In this paper, we take the average of the detection accuracy based on the test set as evaluation metrics.
- **Temporal evaluation.** The performance of temporal evaluation is usually compared to the model training time. To verify the temporal superiority, for each neural, we train 50 epochs based on the same dataset to compare the training time.

Table 2. Comparison of STAPPNP and baseline.

Models	Training time	Detection accuracy
STAPPNP	✓	✓
STGCN(1^{ST} -order approximation) [12]	✓	✓
STGCN(Chebyshev polynomials approximation) [11]	✓	✓
LPSI [14]	×	✓

✓ indicates participation in the corresponding comparison trial and × represents no participation in the corresponding comparison trial, e.g., we compare the detection accuracy but not the training time of LPSI with other models. Since the LPSI algorithm performs rumor source detection in a single snapshot, the input type of LPSI is different from STAPPNP. Therefore, we do not include LPSI in the comparison of training time comparison. In the following, we use STGCN (Cheb) to represent the STGCN (Chebyshev polynomials approximation), and STGCN (1^{ST} -order) represents the STGCN (1^{ST} -order approximation).

Comparison of the Training Time. Table 3 summarizes the results for the training speed of the comparative models. In the experiment, we set the batch size to 50 and compare the training time under the same social network graph. For each neural network model, to ensure input algorithms are the same, we use the one-hot encoding instead of Algorithm 1 as the STAPPNP input algorithm. The results show that the extent of improvement in training speed of STAPPNP increases with the scale of the graph and the complexity of the infection model. STAPPNP achieves the best results in BA networks under both of the two infection models. It improves the training speed by 57s in SI model and 71s in SIR model compare to STGCN (Cheb), which is a significant improvement. STAPPNP also achieves faster training speed than comparing models under all datasets based on two different infection models. Overall, the results show that using APPNP instead of traditional GCN as the spatial layer can improve the training speed under the model significantly. The results demonstrate the effectiveness of our improvements to the ST-Conv block in improving the training speed.

Table 3. Comparisons of training time under two propagation models.

Model	SI				SIR			
	BA	ER	HighSchool	SFHH	BA	ER	HighSchool	SFHH
STGCN (1^{st} -order)	821 s	364 s	289 s	177 s	949 s	670 s	551 s	273 s
STGCN (Cheb)	861 s	369 s	285 s	178 s	951 s	668 s	549 s	275 s
STAPPNP	804 s	353 s	282 s	173 s	880 s	652 s	521 s	267 s

Comparison of Source Detection Accuracy. Since rumor propagation is consistent with the SI infection model, we only verify the superiority of the source detection accuracy under the SI infection model. For the STGCN (1^{st} -order) and STGCN (Cheb), we adopt the one-hot encoding of the node state matrix as input algorithm. LPSI uses 2000 rumor propagation infection graphs without dividing the training set, test set, validation set. The LPSI is not a deep learning model, only deep learning models need to divide the datasets into training, test and validation sets. Thus we use 2000 rumor propagation infection graphs without dividing the training set, test set, validation set and take the last infection result snapshot of each infection graph as LPSI inputs.

As shown in Table 4, which summarizes the result of detection accuracy of deep learning models and LPSI. We divide the test set into four groups and take the average of these four groups as the accuracy result. In the BA network, each deep learning model achieves the best performance. For example, the accuracy of STAPPNP in BA network compared with ER, HighSchool and SFHH networks, rumor source detection accuracy performance increases of 9.5%, 15.5% and 29%, respectively. The results show that for deep learning models, the accuracy of rumor source detection increases with the scale of social networks graph. And we notice that the STGCN (Cheb) outperforms STGCN (1^{st} -order) in a small size graph, but in the large social network BA, STGCN (1^{st} -order) achieves a higher detection accuracy rate than STGCN (Cheb), as stated in [12], the extent of improvement of 1^{ST} -order approximation GCN over chebyshev polynomials approximation GCN increases with the size of the social network graph. Furthermore, to verify that using Algorithm 1 as input algorithm can improve the detection accuracy of STAPPNP, we compare the detection accuracy of STAPPNP to all baselines and find that STAPPNP outperforms all the baseline methods in terms of detection accuracy under all types of networks. STAPPNP achieves the best performance in ER network, the detection accuracy of STAPPNP is 3.5% higher than STGCN (1^{st} -order) in ER network. The results show that taking Algorithm 1 as input algorithm can improve the accuracy of the rumor source detection. The results demonstrate the effectiveness of taking Algorithm 1 as the input algorithm to expresses the relationship between nodes for multiple batches of multiple snapshots.

As shown in Table 4, we notice that STAPPNP significantly outperforms LPSI. For example, in the ER network, the detection accuracy of STAPPNP is 66.45% higher than LPSI. In the BA, High school and SFHH networks, detection accuracy of STAPPNP increases 64.70%, 60.56% and 46.35% over LPSI, respectively. For our experiments, LPSI performs rumor source detection based on a single snapshot and STAPPNP performs rumor source detection based on multiple snapshots. Therefore, we conclude that the better detection performance of STAPPNP is related to much more information obtained from multiple snapshots.

Table 4. Accuracy comparison.

Model	BA	ER	HighSchool	SFHH
LPSI	11.80%	0.55%	0.44%	1.15%
STGCN (1 st -order)	76.00%	63.50%	58.50%	43.50%
STGCN (Cheb)	75.00%	65.00%	59.50%	46.00%
STAPPNP	76.50%	67.00%	61.00%	47.50%

5 Conclusion

In this paper, we investigate the problem of rumor source detection (RSD) under multiple batches of multiple snapshots in the social networks. We propose a modified spatial-temporal convolutional neural network model called STAPPNP. For the RSD problem, we find that taking multiple batches of multiple snapshots as input can improve the detection accuracy. As there is no input algorithm that is suitable for our observation on multiple batches of multiple snapshots, we firstly propose an input algorithm to capture node connectivity features of the multiple batches of multiple snapshots. Our STAPPNP significantly improves the detection accuracy. Meanwhile, to solve the problem of long training time caused by taking the multiple batches of multiple snapshots as input, we improve the ST-Conv block by replacing GCN with APPNP to further increase the training speed. We conduct the experiments on the real world social networks datasets. The results indicate the effectiveness of STAPPNP, which outperforms the 1ST-order approximation STGCN models by about 3.5% in accuracy of rumor source detection, and outperforms the Chebyshev polynomials approximation STGCN 8% in terms of training speed, and even by 66.45% in accuracy of rumor source detection comparing to the LPSI. Since our model is based on the infection model, in the future, we plan to apply STAPPNP in the influence model, filling the gap in the influence model where there are no deep learning models that can aggregate spatio-temporal features.

Acknowledgment. This work is supported in part by the National Natural Science Foundation of China under Grant No. 62202109, and in part by the Guangdong Basic and Applied Basic Research Foundation under Grant No. 2021A1515110321 and No. 2022A1515010611 and in part by Guangzhou Basic and Applied Basic Research Foundation under Grant No. 202201010676.

References

1. Wang, H., Qiao, C., Guo, X., Fang, L., Sha, Y., Gong, Z.: Identifying and evaluating anomalous structural change-based nodes in generalized dynamic social networks. *ACM Trans. Web* **15**(4), 1–22 (2021)
2. Ni, Q., Guo, J., Weili, W., Wang, H., Jigang, W.: Continuous influence-based community partition for social networks. *IEEE Trans. Netw. Sci. Eng.* **9**(3), 1187–1197 (2021)

3. Dong, M., Zheng, B., Quoc Viet Hung, N., Su, H., Li, G.: Multiple rumor source detection with graph convolutional networks. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 569–578 (2019)
4. Wang, Z., Dong, W., Zhang, W., Tan, C.W.: Rumor source detection with multiple observations: fundamental limits and algorithms. *ACM SIGMETRICS Perform. Eval. Rev.* **42**(1), 1–13 (2014)
5. Zhao, Z., Resnick, P., Mei, Q.: Enquiring minds: early detection of rumors in social media from enquiry posts. In: Proceedings of the 24th International Conference on World Wide Web, pp. 1395–1405 (2015)
6. Jin, R., Wu, W.: Schemes of propagation models and source estimators for rumor source detection in online social networks: a short survey of a decade of research. *Discrete Math. Algorithms Appl.* **13**(04), 2130002 (2021)
7. Sha, H., Al Hasan, M., Mohler, G.: Source detection on networks using spatial temporal graph convolutional networks. In: 2021 IEEE 8th International Conference on Data Science and Advanced Analytics, pp. 1–11 (2021)
8. Klicpera, J., Bojchevski, A., Gunnemann, S.: Predict then propagate: graph neural networks meet personalized pagerank. In: Proceedings of the 7th International Conference on Learning Representations (ICLR), pp. 1–15 (2019)
9. Allen, L.J.: Some discrete-time SI, SIR, and SIS epidemic models. *Math. Biosci.* **124**(1), 83–105 (1994)
10. Duvenaud, D.K., et al.: Convolutional networks on graphs for learning molecular fingerprints. In: Advances in Neural Information Processing Systems, vol. 28, no. 9, pp. 2224–2232 (2015)
11. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in Neural Information Processing Systems, vol. 29, pp. 3844–3852 (2016)
12. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: Proceedings of the 5th International Conference on Learning Representations (ICLR), pp. 1–14 (2017)
13. Shah, D., Zaman, T.: Rumors in a network: who’s the culprit? *IEEE Trans. Inf. Theory* **57**(8), 5163–5181 (2011)
14. Wang, Z., Wang, C., Pei, J., Ye, X.: Multiple source detection without knowing the underlying propagation model. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, pp. 217–223 (2017)
15. Li, L., Zhou, J., Jiang, Y.: Propagation source identification of infectious diseases with graph convolutional networks. *J. Biomed. Inform.* **116**, 103720 (2021)
16. Shah, D., Zaman, T.: Rumor centrality: a universal source detector. In: Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE Joint International Conference on Measurement and Modeling of Computer Systems, vol. 40, pp. 199–210 (2012)
17. Albert, R., Barabasi, A.L.: Statistical mechanics of complex networks. *Rev. Mod. Phys.* **74**(1), 47–92 (2002)
18. Salathe, M., Kazandjieva, M., Lee, J.W., Levis, P., Feldman, M.W., Jones, J.H.: A high-resolution human contact network for infectious disease transmission. *Proc. Nat. Acad. Sci.* **107**(51), 22020–22025 (2010)
19. Genois, M., Barrat, A.: Can co-location be used as a proxy for face-to-face contacts? *EPJ Data Sci.* **7**(1), 1–18 (2018). <https://doi.org/10.1140/epjds/s13688-018-0140-1>