Predicting Information Diffusion Cascades Using Graph Attention Networks

Meng Wang and Kan Li(✉)

School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China
{3120181036,likan}@bit.edu.cn

Abstract. Effective information cascade prediction plays a very important role in suppressing the spread of rumors in social networks and providing accurate social recommendations on social platforms. This paper improves existing models and proposes an end-to-end deep learning method called CasGAT. The method of graph attention network is designed to optimize the processing of large networks. After that, we only need to pay attention to the characteristics of neighbor nodes. Our approach greatly reduces the processing complexity of the model. We use realistic datasets to demonstrate the effectiveness of the model and compare the improved model with three baselines. Extensive results demonstrate that our model outperformed the three baselines in the prediction accuracy.

Keywords: Social network · Information cascade prediction · Graph attention network

1 Introduction

The emergence of online social platforms, such as Twitter, Weibo, Facebook, Instagram, WeChat, QQ, has changed our daily lives. Information diffusion cascade occurs when people see the tweets of others on the platforms and then retweet the information that the others have written. Cascade information can play an important role in addressing the issue of controlling and predicting information diffusion. Along with this growth of the messages on these platforms, however, there are increasing concerns over complex and dynamic information diffusion cascade prediction.

Much work so far has focused on information diffusion cascade prediction. It mainly falls into four main categories: feature-based method, generative process method and deep learning-based methods. Information diffusion cascade prediction still has many challenges: (1) lack of knowledge of dynamic network graph structure when real social networks are constantly changing. This greatly affects the general type of the model. (2) the accuracy of the information diffusion cascade prediction take into account not only structural information but also temporal information.

H. Yang et al. (Eds.): ICONIP 2020, CCIS 1332, pp. 104–112, 2020.
https://doi.org/10.1007/978-3-030-63820-7_12

Here, we introduce an attention and temporal model called CasGAT to predict the information diffusion cascade, which can handle network structure predictions in different time periods. It only deals with the relationship between nodes and neighbor nodes in the graph, without first acquiring the structural features of the entire graph. Therefore, it can handle tasks including evaluating models on completely invisible graphs during training. In addition, CasGAT uses deep learning methods, graphical attention networks and time decay to enhance our framework.

In the rest of this paper, Sect. 2 reviews the related work. In Sect. 3, we describe the main aspects of CasGAT methodology in details. Experimental evaluations quantifying the benefits of our approach are presented in Sect. 4 and Sect. 5 concludes the paper and outlines directions for future work.

2 Related Work

In this section, we briefly review the research on the Information diffusion cascade prediction. In general, existing method of information diffusion cascade prediction falls into three main categories.

2.1 Feature-Based Method

As for early adopts and network structure, the properties of featured-based methods energetically depends on the quality of the hand-crafted features. These features are mainly extracted from temporal features [8,10], structural features [1,12], content features [5,11] and features defined by early adopters [4,6]. Faced with complex problems and massive data, it is difficult for people to systematically design and measure Effectively capture complex features of relevant information.

2.2 Generation Process Method

The generation process method focuses on modeling the strength function independently for each message arrival. Generally, they observe each event and learn the parameters by maximizing the probability of the event occurring within the observation time window. Shen et al. [9] used the enhanced Poisson process to model three factors in the social network (the influence of nodes, the decay process of information heat, and the "rich get rich" mechanism). These methods show enhanced interpretability, but the implicit information in cascade dynamics cannot be fully utilized to obtain satisfactory predictions.

2.3 Deep Learning-Based Method

Deep learning-based methods are inspired by the latest success of deep learning in many fields, and cascade prediction using deep neural networks has achieved significant performance improvements. The first proposed information cascade

predictor (DeepCas) based on deep learning [7] converts a cascade graph into a sequence of nodes by random walk, and automatically learns the representation of each graph. Cao et al. [2] proposed a process based on deep learning, which inherits the high interpretability of the Hawkes process and has the high predictive power of deep learning methods. However, due to the low efficiency of cascading sampling bias and local structure embedding, they lack good learning ability in cascading structure information and dynamic modeling.

3 Model

The information cascade is affected by many factors. The overview of the architecture is shown in Fig. 1. The model takes a cascade graph as the input and predicts the information increment size as the outputs. Next, we focus on the detailed methods one by one.

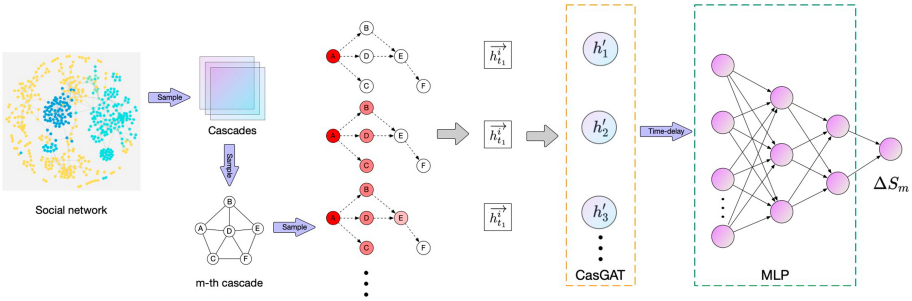

Fig. 1. The overview of our model architecture.

3.1 Basic Definitions

We denote a social network (e.g., weibo network) as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of all users of $\mathcal{G}$, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of all relationships between users. Suppose we have M pieces of message in the social network, let $C = \{C_m, (1 \leq m \leq M)\}$ be the set of M information cascades. We use $g_{t_i}^i = \left(v_{t_i}^i, e_{t_i}^i\right)$to represent the cascade path for the i-th information diffusion of the message m at t_i, where $v_{t_i}^i$ represents the subset of V, $e_{t_i}^i$ represents the relationship between the users in $v_{t_i}^i$. The objective problem of this paper can be expressed as: given a set of information cascades $C_m(t)$ observed on a given social network information cascade $C = (C_1, C_2, \ldots, C_M)$, we want to predict the increment size ΔS_m of each information cascade C_m for a fixed time window Δt, i.e., $\Delta S_m = \left|S_m^{T+\Delta t}\right| - \left|S_m^T\right|$.

3.2 Cascade Embedding

Each information cascade on every social media almost consists of a sequence of retweets, so the big cascade is becoming a big problem when processing the original graph cascade. There are different ways to represent cascaded input. In our approach, the cascade C_m represents the propagation path of the message m. We can get the cascade graph $G^i_{t_i}$ which is represented as a set of cascade paths that are sampled through the $G^i_{t_i} = \left\{g^i_{t_1}, g^i_{t_2}, g^i_{t_3}, \ldots, g^i_T\right\}, t_i \in [0, T)$. Each cascade graph not only contains the structure information between the users, but also carries the temporal information. $G^i_{t_i}$ consists of a set of matrices.

3.3 Graph Attention Layer

This part of our model is based on graph attention network. In our approach, we use the method called feature decomposition to compute the eigenvectors of the above matrix in $G^i_{t_i}$. And we get the set of node features, $\mathbf{h} = \{\overrightarrow{h^i_{t_1}}, \overrightarrow{h^i_{t_2}}, \overrightarrow{h^i_{t_3}}, \ldots, \overrightarrow{h^i_T}\}$ as the input of graph attention layer. In order to get the corresponding conversion between the input and output, we need to perform linear transformation at least once according to the input features to get the output features, so we need to train a weight matrix $\mathbf{W}$ for all bytes. The attention coefficients

$$e_{uv} = f(W\overrightarrow{h^i_u}, W\overrightarrow{h^i_v}) \tag{1}$$

indicate the importance of node v's features to node u. In order to make the attention coefficient easier to calculate and compare, we introduced softmax function to regularize all adjacent nodes of u:

$$\alpha_{uv} = softmax_v\left(e_{uv}\right) = \frac{\exp\left(e_{uv}\right)}{\sum_{k\in V}\exp\left(e_{uk}\right)} \tag{2}$$

Combining the above formulas (1) and (2), sorting them together can get the complete attention mechanism as follows:

$$\alpha_{uv} = softmax_v\left(e_{uv}\right) = \frac{\exp\left(LeakyReLU\left(\overrightarrow{f}^T\left[W\overrightarrow{h^i_u} \| W\overrightarrow{h^i_v}\right]\right)\right)}{\sum_{k\in V}\exp\left(LeakyReLU\left(\overrightarrow{f}^T\left[W\overrightarrow{h^i_u} \| W\overrightarrow{h^i_k}\right]\right)\right)} \tag{3}$$

where $.^T$ represents transposition and $\|$ is the concatenation operation

The outputs of this layer are represented:

$$h_{struc} = \boldsymbol{h} = \sigma\left(\sum_{v\in V}\alpha_{uv}\overrightarrow{h^i_v}\right) \tag{4}$$

The output characteristics of this node are related to all the nodes adjacent to it, which are obtained after their linear and nonlinear activation.

3.4 Temporal Embedding

In our model, we will embed time. We use the most common LSTM (Long short-term memory) to model temporal embedding. Below we will talk about the specific implementation of these aspects. LSTM is a special kind of RNN. We use the most common LSTM (Long Short Term Memory) to model temporal embeddings. LSTM is a special RNN that can selectively memorize past information through a memory unit. The specific implementation method is as follows:

$$\begin{aligned} f_t &= \sigma\left(W_f \cdot \left[h_{t-1}, g_t^i\right] + b_f\right) \\ i_t &= \sigma\left(W_i \cdot \left[h_{t-1}, g_t^i\right] + b_i\right) \\ \tilde{C}_t &= \tanh\left(W_C \cdot \left[h_{t-1}, g_t^i\right] + b_C\right) \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\ o_t &= \sigma\left(W_o \left[h_{t-1}, g_t^i\right] + b_o\right) \end{aligned} \tag{5}$$

where g_t^i is a matrix representation of the cascade path, and f_t, i_t, o_t are respectively the input gate, forget gate and output gate.The hidden state is then updated by:

$$h_t = o_t * \tanh\left(C_t\right) \tag{6}$$

Finally, we get the collection of the m messages $h_{tem} = h = \{\overrightarrow{h_1}, \overrightarrow{h_2}, \overrightarrow{h_3}, \ldots, \overrightarrow{h_M}\}$.

3.5 Time-Delay Function and Output Layer

Social networking is a very dynamic process. For example, when a Weibo has just been published, it has the greatest influence, and then its effect will become smaller and smaller over time. This is why we have to add the time decay factor. Current methods have power-law functions ($\phi(t) = (t+c)^{-(1+\theta)}$) and exponential functions ($\phi(t) = e^{-\theta t}$) to simulate. In our model, we use a non-parametric method to make a more appropriate representation of the time decay factor. We assume that the overall observation time window is $[0, T)$, and then split the observation time window into l disjoint time intervals $\{[t_0 = 0, t_1), [t_1, t_2), \ldots [t_{l-1}, t_l = T)\}$. We need to learn the discrete variable of time decay effect $\lambda_j, j \in (1, 2, \ldots, l)$. For different cascade paths, we all add time decay parameters to get a new representation:

$$h_t{}' = \lambda_m(h_{tem} + h_{struc}) \tag{7}$$

and the m-th cascade path can be represented by:

$$h'\left(C_m\right) = \sum_{t=1}^{T} h_t' \tag{8}$$

The final output of our model consists of a multi-layer perceptron (MLP), expressed as follows:

$$\Delta S_i = \mathbf{MLP}\left(h'\left(C_i\right)\right) \tag{9}$$

4 Experiments

In this section, we will present the details of experiments conducted on real-world datasets and the results analysis between our proposed model and baseline.

4.1 Dataset

The dataset is from Sina Weibo, one of the most popular microblogging platform in China. This dataset is provided by Cao [2]. In Sina Weibo, the network is composed of the relationship among a big group of users. The relationship contains the retweets, likes and comments. In this paper, we concentrate on the retweets of messages. Compared to likes and comments, retweet is the most direct way to form a cascade. The crawl time of dataset is from 0:00 on June 1, 2016 to 24:00 on June 1, 2018. It remains 119,313 messages in total. We filter out the message before 6 am and after 9 pm. And the length T of observation time window, we consider three settings, i.e., T = 1 h, 2 h and 3 h. Finally, we sort all the rest cascades by publishing time, replacing the first 70% as the training set, the middle 15% as the verification set, and the last 15% as the test set.

4.2 Baselines and Variants of Our Model

For a comprehensive comparison, we considered a variety of the latest alternatives in the methods mentioned above. The baselines are DeepCas [7], DeepHawkes [2] and CasCN [3]. In addition to comparing with existing baseline, we also compared with some variants of CasGAT. Here are a few variants of our model: CasGAT-GCN, CasGAT-GRU, CasGAT-Time.

4.3 Evaluation Metric

For the information cascade prediction problem, we choose standard evaluation metrics—MSLE (mean square log-transformed error) in our experiments. The smaller MSLE, the better prediction performance. Specifically, MSLE is the metric for evaluating the linking accuracy, defined as:

$$MSLE = \frac{1}{M}\sum_{i=1}^{M}\left(\log \Delta S_i - \log \Delta \widetilde{S}_i\right)^2 \tag{10}$$

where M is the total number of the messages, ΔS_i is the prediction result and $\Delta \widetilde{S}_i$ is the actual result.

mSLE is the median of $\left(\log \Delta S_i - \log \Delta \widetilde{S}_i\right)^2$ which can effectively reduce the impact of outliers, defined as:

$$mSLE = median_{i=1\ldots M}\left(\log \Delta S_i - \log \Delta \widetilde{S}_i\right)^2 \tag{11}$$

Table 1. The performance of baseline model and CasGAT on Sina Weibo dataset

Observation Time	1 h		2 h		3 h	
Evaluation Metric	mSLE	MSLE	mSLE	MSLE	mSLE	MSLE
DeepCas	0.918	3.693	0.857	3.276	0.906	3.212
DeepHawkes	0.736	2.501	0.689	2.384	0.694	2.275
CasCN	0.638	2.375	0.615	2.243	0.542	2.098
CasGAT	**0.606**	**2.253**	**0.547**	**2.093**	**0.503**	**1.936**

4.4 Result

In this section, we compare the performance of our model with the three baselines. The results are illustrated in Table 1.

Table 1 summarizes the performance comparison between CasGAT and three baselines on the Sina Weibo dataset. Generally speaking, the proposed CasGAT model performs relatively well in the information cascade prediction on the public Sina Weibo dataset. It is superior to traditional methods, such as feature-based methods and generation methods, and superior to the latest deep learning methods. MSLE and mSLE are statistically significantly reduced.

To study and prove the effectiveness of each component of our model, we introduced three variants of CasGAT. These variants are all modified models for a part of our model framework. The experimental results are shown in Table 2, from which we can see that the original CasGAT caused a certain reduction in prediction error compared to other variants. Through comparison with CasGAT-Time, we find that the time decay effect is essential for cascade size prediction. Similarly, CasGAT-GCN and CasGAT-GRU also reduce prediction performance to some extent. Among them, the error of CasGAT-GRU variant is smaller than the original model within 1 h, but the error of the original model is still small in the subsequent time.

Table 2. The performance of CasGAT and its variants on Sina Weibo dataset

Observation time	1 h		2 h		3 h	
Evaluation Metric	mSLE	MSLE	mSLE	MSLE	mSLE	MSLE
CasGAT-GCN	0.618	2.292	0.574	2.186	0.525	1.994
CasGAT-GRU	**0.601**	**2.249**	0.550	2.138	0.516	1.952
CasGAT-Time	0.927	2.641	0.845	2.598	0.701	2.336
CasGAT	0.606	2.253	**0.547**	**2.093**	**0.503**	**1.936**

In summary, structural and time information are two key components in CasGAT. These two factors play an indispensable role in improving prediction accuracy.

5 Conclusion

We propose a new information cascade propagation model based on deep learning - CasGAT. Our model uses structure and temporal information to achieve information cascade prediction. The model mainly adds a graph attention mechanism, which greatly reduces the complexity of modeling graphs. We put our focus on neighbor nodes instead of the entire graph, which can further improve our prediction performance. In the future, we will increase our efforts to expand the model's perception capabilities and extend the model to more effective data sets.

Acknowledgments. This research was supported by Beijing Natural Science Foundation (No. L181010, 4172054), National Key R & D Program of China (No. 2016YFB0801100), and National Basic Research Program of China (No. 2013CB329605).

References

1. Bao, P., Shen, H.W., Huang, J., Cheng, X.Q.: Popularity prediction in microblogging network: a case study on Sina Weibo. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 177–178 (2013)
2. Cao, Q., Shen, H., Cen, K., Ouyang, W., Cheng, X.: DeepHawkes: bridging the gap between prediction and understanding of information cascades. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pp. 1149–1158 (2017)
3. Chen, X., Zhou, F., Zhang, K., Trajcevski, G., Zhong, T., Zhang, F.: Information diffusion prediction via recurrent cascades convolution. In: 2019 IEEE 35th International Conference on Data Engineering (ICDE), pp. 770–781. IEEE (2019)
4. Cui, P., Jin, S., Yu, L., Wang, F., Zhu, W., Yang, S.: Cascading outbreak prediction in networks: a data-driven approach. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 901–909 (2013)
5. Hong, L., Dan, O., Davison, B.D.: Predicting popular messages in Twitter. In: Proceedings of the 20th International Conference Companion on World Wide Web, pp. 57–58 (2011)
6. Lerman, K., Galstyan, A.: Analysis of social voting patterns on digg. In: Proceedings of the First Workshop on Online Social Networks, pp. 7–12 (2008)
7. Li, C., Ma, J., Guo, X., Mei, Q.: DeepCas: an end-to-end predictor of information cascades. In: Proceedings of the 26th International Conference on World Wide Web, pp. 577–586 (2017)
8. Pinto, H., Almeida, J.M., Gonçalves, M.A.: Using early view patterns to predict the popularity of YouTube videos. In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, pp. 365–374 (2013)
9. Shen, H., Wang, D., Song, C., Barabási, A.L.: Modeling and predicting popularity dynamics via reinforced Poisson processes. In: Twenty-Eighth AAAI Conference on Artificial Intelligence (2014)
10. Szabo, G., Huberman, B.A.: Predicting the popularity of online content. Commun. ACM **53**(8), 80–88 (2010)

11. Tsur, O., Rappoport, A.: What's in a hashtag? Content based prediction of the spread of ideas in microblogging communities. In: Proceedings of the fifth ACM International Conference on Web Search and Data Mining, pp. 643–652 (2012)
12. Weng, L., Menczer, F., Ahn, Y.Y.: Predicting successful memes using network and community structure. In: Eighth International AAAI Conference on Weblogs and Social Media (2014)