An Attention-based Neural Popularity Prediction Model for Social Media Events

Guandan Chen[1,2], Qingchao Kong[1,2], Wenji Mao[1,2]
[1]The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation,
Chinese Academy of Sciences, China
[2]School of Computer and Control Engineering, University of Chinese Academy of Sciences, China
{chenguandan2014, qingchao.kong, wenji.mao}@ia.ac.cn

***Abstract*—Online interaction behavior between web users often makes some events go viral. Popularity prediction of events is a key task in many security related applications. It forecasts how widely events would spread based on the information of evolution at an early stage. Existing methods either rely on careful feature engineering, or solely consider time series, ignoring rich information of user and text content. In this paper, we attempt to extract and fuse the rich information of text content, user and time series in a data-driven fashion. To this end, we design a popularity prediction model based on deep neural networks, which uses three encoders to extract high-level representation of text content, users and time series respectively. In addition, we incorporate attention mechanism to make our model focus on important features. Experiments on real world dataset show the effectiveness of our proposed model.**

Keywords—popularity prediction; deep neural network; attention mechanism

I. Introduction

The development of social media makes interactions between people more easily and thus some events can spread through the Internet more quickly than ever before. The social media platform can be an opportunity or a disaster for public management, business at the same time. For example, public opinion of can make a company become famous overnight, or make the stock price of a company decline sharply. Moreover, the spread of controversial events can cause misunderstanding and distrust between people and government. Thus, analyzing the popularity spread of social media events has become a critical task for many security-related applications.

Predicting the popularity of social media events can help measure the impact of threats and the need of citizens [1]. It can also help better decision making by knowing the emergency events about natural disasters, terrorisms, crimes, and many others. However, popularity prediction is not a trivial task. Firstly, the amount of users and tweets in social media is huge. Secondly, there are many informal expressions, word morphs in social media, which makes the data very noisy. Thirdly, there are missing data on the social media platform.

To address the aforementioned challenges, in this paper, we propose a deep neural network model to predict the popularity of social media events, which jointly models the users engaging the events, text contents of the events as well as popularity evolving information at an early stage. Our method maps words and users into a lower dense space by considering word context and the structure of user graph respectively. It then extracts high-level representation of features automatically using a multi-layer neural network. To make the model focus on the most important features, the attention mechanism is also introduced in our proposed model.

The contributions of this paper are as follows: (1) we propose a neural network based popularity prediction model that considers text content, user and popularity time series information for social media events; (2) to make the model focus on the most important features, we introduce the attention mechanism in our proposed model to further improve the prediction performance; (3) we use a large social media event dataset to show the effectiveness of our proposed popularity prediction model.

II. Related Work

One branch of popularity prediction method focuses on feature design. [2] extracts topics, user and time series features, and uses different machine learning model to predict the topic popularity in the future days. [3] predicts the popularity of hashtags at next day, using features include the divergence of language models of the event and all tweets at that day, 20 dimension topic distribution et.al. A major drawback of these methods is they usually require domain knowledge and careful engineering to design useful features for prediction.

Another branch seeks for explaining the formation of the popularity. [4] simulates the spreading process of the events, which models user interest by topic distribution of their tweets and uses cosine similarity between topic distributions of users and events to compute the interest of users towards the events. Much related work models the spread of the content as a probabilistic process [5]. [6] is state of the art of these methods. It uses Hawkes process to model the spreading process of events, then uses both learned parameters and other features to train a random forest model. These above models elegantly explain the formation process of popularity, however, most of these works only use time series information, and ignore the user and text content information, except for [6] using a hybrid method.

Deep neural networks (DNN) has beaten records in many tasks, such as computer vision and speech recognition[7]. Motivated by the successful application of DNN in the above fields, we propose an attention-based neural network model to learn high-level representations for text content, user and popularity time series information and combine these representations as features for popularity prediction.

III. Attention-based Neural Popularity Prediction Method

A. Problem Formulation

We define popularity as the number of tweets related to an event. Considering most applications of popularity prediction do not care about the accurate value of popularity, we transform the popularity prediction task to predicting whether the future popularity of an event will exceed a given threshold. Specifically, given an event H, we observe the event during time period $[T_s, T_s + t_o]$, including related tweets, authors of the tweets as well as the publication timestamps of the tweets, where T_s is the publication time of the first tweet related to the event, t_o measures how long we observe. Denote the number of tweets about event H from T_s to $T_s + t_r$ as V, where t_r represents how long we want to predict. Our goal is to predict whether V will exceed threshold δ.

B. Neural Popularity Prediction Model

Our neural popularity prediction model is composed of four parts, namely, text content encoder, user encoder, time series encoder and the fusing layer. The above three encoders are to learn text, user and time series representation from data respectively, while the fusing layer combines features produced by these three encoders, and outputs prediction results.

Text content encoder. We use a hierarchical neural network to obtain word, tweet and event representations. At word level, we embed each word w_j^i in a tweet into a low dimension vector x_j^i using glove[8], which is a widely used word embedding model. Then we use bidirectional Gated Recurrent Units (GRU) to get representation of words. A bidirectional GRU map a sequence to another sequence, i.e. $[h_1^i, h_2^i, \dots, h_n^i] = BiGRU([x_1^i, x_2^i, \dots, x_n^i])$, where each h_j^i summarizes the context information of word w_j^i. To get the representation of a tweet, we introduce the attention mechanism[9] to extract and aggregate the most import states for popularity prediction. This operation produce one single vector by weighted summing the sequence of vectors, which are denoted as $s_i = Att([h_1^i, h_2^i, \dots, h_n^i])$. For the event representation, we first extract representations of each tweet related to the target event, then use bidirectional GRU and the attention mechanism to get the event representation: $v_c = Att(BiGRU([s_1, s_2, \dots, s_n]))$.

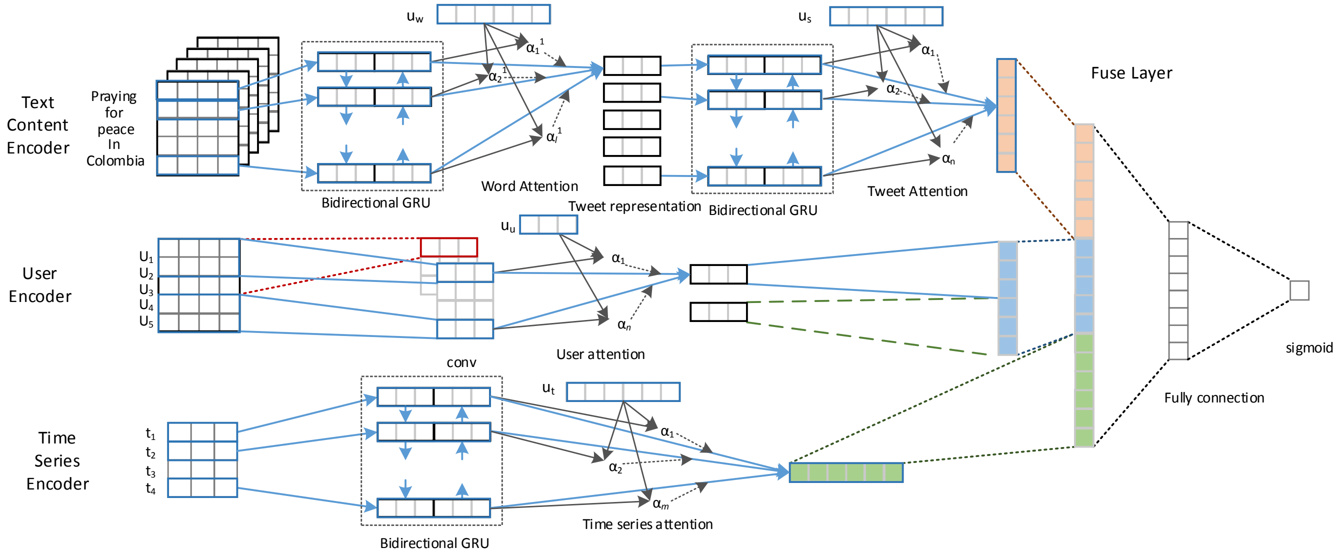
User encoder. Each user U_i is represented as a low dimension vector using node2vec[10], which encodes the structure and community information of the users in the network. Then we use convolutional neural network (CNN) and an attention layer to get a higher level representation of users. Specifically, we aggregate the user representation in each set of filters by introducing attention mechanism $v = Att(CNN([U_1, U_2, \dots, U_n]))$. K sets of filters would produce K vectors $v_1, v_2, \dots v_K$ and these K vectors are combined into one vector $v_u = [v_1, v_2, \dots, v_k]$.

Time series encoder. We divide the early observation time $[T_s, T_s + t_o]$ into m time windows with fixed length Δt. Each time widow corrsponds to one feature vector, denoted as f_i. The feature vector includes number of tweets at this time window, average and maximum follower number of authors of these tweets. Thus the whole time series is represented as $[f_1, f_2, \dots, f_m]$. We again use the bidirectional GRU and attention mechanism to map this sequence of feature vector to the representation of time series, i.e. $v_t = Att(BiGRU([f_1, f_2, \dots, f_m]))$.

Fusing layer. Above encoded text content, users and time series representations are concatenated into one feature vector after batch normalization[11]. A dense layer would encode the feature further. Then a sigmoid function is used to produce popularity prediction result.

C. Optimization

We use cross-entropy as our loss function, i.e. $L = -\sum_{i=1}^{M}(t_i \, log \, p_i + (1 - t_i) \, log(1 - p_i))$. Here, t_i and p_i is the true label and predict result of i-th sample respectively. We adopt Adagrad to optimize the model, which is a widely used optimization method.



IV. Experiment Result

A. Dataset and Experiment Setting

The dataset was collected using Twitter public API[1] from Aug. 9, 2016 to Dec. 10, 2016. Users usually use a hashtag to denote the event they discuss, but sometimes one event may correspond to several hashtags. Thus, we merge the hashtags that differ in case only and take every merged hashtag as an event. The observe time t_o is set to 1, 6, 12 or 24 hours, and prediction time is 7 days. The top 20% most popular events are considered as popular events, and we randomly sample the same number of events from the rest events as unpopular events. The training set contains 80% events in first three months, and the validation set contains the other 20% events, and test set contains each event at the last month. We use accuracy as our metric.

TABLE I. Summary statistics of the dataset

Attributes	#events	#tweets	avg. #tweets per event	max #tweets per event
Statistics	117,603	38,255,917	325	234,944

B. Baseline methods

We compare our method to the following baseline methods using our dataset. (1) *M-1* [2]: It mainly uses lexicon features, associated with a few user, user interaction and time series features to predict popularity. (2) *M-2* [3]: It proposes many novel features based on language model, network structure and time series to predict popularity. (3) *Hawkes* [6]: It models popularity dynamics as Hawk process, then combine Hawk process parameters, a few user and text content features to predict popularity.

The following variations of our model are also taken as baseline methods. (1) *TCE* (Text Content Encoder): It only uses features produced by text content encoder to predict popularity. (2) *UE* (User Encoder): It only uses features produced by user encoder to predict popularity. (3) *TSE* (Time Series Encoder): It only uses features produced by time series encoder to predict popularity. (4) *NPP-NA*: It removes attention mechanism from our neural popularity prediction model.

C. Results

Table II shows the experimental results of the above baseline methods and our neural popularity prediction model. We can see that our neural popularity prediction model outperforms all the baseline methods. All models have lower accuracy when the observed time is short because there is little information for popularity prediction. The time series encoder has the lowest accuracy when the observed time is 1 hour. It is because that we separate observed time t_o into fixed length time windows, which may lose too much information when observed time is very short. Our text content encoder, time series encoder has better accuracy than M-1, M-2 and Hawkes when the observed time is 12 or 24 hours. It shows that our encoders has powerful representation ability. When there exists enough information, they can outperform models using rich features. Combining three encoders, we can get a 2-3% accuracy improvement. By adding attention mechanism, we can further improve the accuracy, which shows the power of attention mechanism.

TABLE II. Accuracy of different methods

Methods	**Observed Time(hours)**			
	1	***6***	***12***	***24***
M-1	0.612	0.653	0.682	0.722
M-2	0.633	0.701	0.728	0.781
Hawkes	0.590	0.669	0.709	0.749
TCE	0.618	0.687	0.731	0.786
UE	0.607	0.685	0.727	0.779
TSE	0.507	0.679	0.721	0.793
NPP-NA	0.615	0.704	0.755	0.809
NPP	**0.651**	**0.723**	**0.764**	**0.818**

V. Conclusion

In this work, we propose an attention-based neural popularity prediction model for social media events. The proposed method learns high-level representations of text content, users and time series in a data-driven approach. Also, the attention mechanism is introduced to make the model focus on the important features for popularity prediction. For evaluation, we conduct experiments using Twitter dataset, and experimental results show that our method outperforms the baseline methods.

Acknowledgement

This work is supported in part by NSFC Grant #71621002, #61671450 and #71472175, and the Ministry of Science and Technology of China Major Grant.

References

[1] Q. Kong, W. Mao, D. Zeng, and L. Wang, "Predicting popularity of forum threads for public events security," in *Intelligence and Security Informatics Conference (JISIC), 2014 IEEE Joint*, 2014, pp. 99-106: IEEE.

[2] O. Tsur and A. Rappoport, "What's in a hashtag?: content based prediction of the spread of ideas in microblogging communities," in *WSDM*, 2012, pp. 643-652: ACM.

[3] L. M. Aiello *et al.*, "Sensing trending topics in Twitter," *IEEE Transactions on Multimedia,* vol. 15, no. 6, pp. 1268-1282, 2013.

[4] X. Zhang, X. Chen, Y. Chen, S. Wang, Z. Li, and J. Xia, "Event detection and popularity prediction in microblogging," *Neurocomputing,* vol. 149, pp. 1469-1480, 2015.

[5] P. Bao, H.-W. Shen, X. Jin, and X.-Q. Cheng, "Modeling and predicting popularity dynamics of microblogs using self-excited hawkes processes," in *WWW*, 2015, pp. 9-10: ACM.

[6] S. Mishra, M.-A. Rizoiu, and L. Xie, "Feature driven and point process approaches for popularity prediction," in CIKM, 2016, pp. 1069-1078: ACM.

[7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature,* vol. 521, no. 7553, pp. 436-444, 2015.

[8] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global Vectors for Word Representation," in *EMNLP*, 2014, vol. 14, pp. 1532-1543.

[9] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of NAACL-HLT*, 2016, pp. 1480-1489.

[10] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *SIGIR*, 2016, pp. 855-864: ACM.

[11] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *ICML*, 2015, pp. 448-456.

[1] https://dev.twitter.com/streaming/overview