Graph Neural Networks in Analysis of Main Influencing Factors and Modeling of Network Content Propagation Rule

Yupei Li

November 14, 2022

Graph Neural Networks in Analysis of Main Influencing Factors and Modeling of Network Content Propagation Rule*

Yupei Li

Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an, China

Abstract

In recent years, various social media platforms have come into being, and the information diffusion through social networks has become more frequent and complex. Social network information dissemination rules can be used to quickly match the relevant user groups with information to achieve faster speed of news dissemination. Research on the rules for dissemination of information and its influencial factors have become significant in field of networks. Further, it is necessary to regulate harmful information and also provide recommendations. Therefore, it is important to predict patterns of information diffusion in social networks, and improve recommendations and supervision by accurately identifying the object (message receiver) of message transmission and whether the object (message sender now) forwards the message to cause another message diffusion.

Based on the graph neural network, this paper proposes an ICGNN model, studies the factors that affect information diffusion, and predicts whether a blog will eventually be forwarded to other users (message receiver) by the message sender. It also analyses the factors that affect the rule of message transmission, and conducts a comparison of these factors with case analysis. During the whole experiments, firstly, this paper uses the public microblog data set, and collects and analyzes user characteristics, network structure factors and information content affecting information diffusion. Secondly, Jieba word segmentation and Bert are used to process text features, and node features are obtained by filtering incomplete nodes, achieving graph embedding. Additionally, the independent cascading model is reproduced and trained to converge. Assuming that the message propagation is only determined by the graph structure, the probability of message passing through each edge is obtained and taken as the weight of the edge. Finally, a multi-label classification model was designed. Then the sender of the message is used as a label and encoded with one hot vector as a label, which is dimensionality reduced by an encoding and decoding model. After that, features were set into a graph convolutional network and graph attentional network for comparative experiments, and then labels were restored by threshold for model evaluation.

Through the verification and the testing of the model, the recall rate is 52% and the accuracy rate is up to 69%. The results show that the ICGNN model has good ability to predict the pattern of information diffusion, and thus can be popularized in future research. Moreover, a prediction result is selected for case analysis, compare the information and influencing factors of the message sender and receiver, summarize the influence degree of some influencing factors, and put forward future research direction.

Keywords: Graph neural network; Natural Language Processing; Social networking; Information dissemination; Independent cascade model; Multi-label classification

1 Introduction

In the research on information cascade diffusion, Yang et al. (2018)put forward the NDM neural diffusion model, defined the criteria for infecting users, combined in-depth learning with network diffusion for the first time, and applied the model to four data sets based on a relaxation assumption, to solve the key point that the complex model could not simulate the real complex network transmission process before, and in practice, it was concluded that it had good robustness. However, the key to simulate the real process is that the input of timing propagation is required. Liu, Bao, Zhang, Tang, and Xiong (2020) designed a three-layer model with attention and multi-layer perceptron based on solving the problem of time characteristics of cascade prediction, and applied it to multiple data sets such as Twitter for verification. Combined with dynamic modeling, the model

*This essay is part of the undergraduate dissertation essay due to the characteristics limitation has a small error. These gave an instinct on how information diffusion model constructed and its focused problem.

Wu et al. (2020) reviewed the underlying principles and application advantages of existing graph neural networks, and analyzed four kinds of GNN: the graph attention network, graph automatic encoder, graph generation network, and graph space-time network, promoting the research of graph neural networks. Murphy, Laurence, and Allard (2020) applied the combination of MLP, GNN models and in-depth learning to the network propagation prediction of COVID-19, achieved good results in Spain. Ding, Yan, Zhang, Dai, and Dong (2016) proposed to use semi-supervised learning based on graph neural networks for deep learning, tackle the issue with predicting hidden attributes of users by using graph structures and user known attributes, define the weight relationship of edge connections among users, and provide further research plans for SAN models. Michalski, Kazienko, and Król (2012) used a variety of machine learning methods and a variety of measurements to predict the social public opinion dataset of time series data, compared the conclusions, and concluded that the adaptability of their various learning methods was not ideal. Hu et al. (2015) proposed to use TDF parallel computing to simulate real medical scenarios by modeling social network points, and the results were applied to the spread prediction of avian influenza in public health emergencies. Mashhadi, Mokhtar, and Capra (2009) proposed the Habit delay tolerant network DTN to enable relevant content to reach the intermediate node of interest, and proposed a new network propagation mode to reduce overhead without affecting performance. These researches involving exsiting models related to Graph Neural Networks gave many options that could be chosen in my own model.

Guille, Hacid, Favre, and Zighed (2013) reviewed the information dissemination model, divided it into an interpretative graph and a predictive non graph, and discussed the technical focused contents and challenges of each model. Jiuxin et al. (2014) analyzed and integrated features that affect information dissemination, such as the number of user fans, forwarding activity, etc., and verified the accuracy of these features with machine learning methods such as Bayesian Inference.

Yangfan (2021) used natural language processing to detect false information content and extract information features.Jiao (2019) used long and short memory neural network modeling and based on the good application of natural language processing in convolutional neural networks, she proposed the research on false content detection of information dissemination oriented to microblog, and improved the existing algorithm. These inspired the section of preprocessing the textual informaton collected in blog dataset.

The above papers either focus on the improvement and optimization of a graph neural network and its deep learning model, or develop a new cascade model, or summarize the characteristics of social network information dissemination. It can be concluded that graph neural networks and deep learning models can be used for information transmission, but there is still a lack of practical application of high-performance and universal verification sets in social neural networks. The main focus of research has been on model development, or on information dissemination applications. This study further enriches integration of things above in model related to graph neural network and deep learning, compares the performance of corresponding model applications in social network information dissemination, extracts the features that have an important impact on social network information dissemination, and makes contributions to the use and supervision of social networks.

2 Problem Definition

Under the collected public microblog data set, each user has 11 user characteristics, and all users with their characteristics constitute a total set of users. There are message senders U_s, who send blog posts containing text messages, and then spread them to message receivers U_r, which are defined as users who forward blog posts. The main question in this study is defined as predicting the message receiver of each blog post, specifically predicting the accuracy of each one-dimensional label of each user. Here, the label l represents whether U_r is receiving messages from U_s.

In this paper, we consider the influence of user characteristics, information characteristics and graph structure characteristics of the rule of information dissemination. However, this only predicts whether the message will be received by U_r correspondingly, without considering the time factor and the order of forwarding articles in the process. Further, this paper applies the model to the public microblogging data set for testing the whole model.

3 Model

The whole model is divided into the following sections: First, the paper combs the components of the model and analyzes the contents of each part, from the inter independent cascade model to multi-label classification design to the final graph neural network, and analyzes the correlation points in detail; Next, I will collect and analyz the factors that affect information dissemination; Ater that, I will collect and sort out the public microblog data set. Data preprocessing is used to obtain user U centered non-text features. The Jieba word segmentation and

Bert model are used to process text features, and self-coding dimension reduction is used to integrate user's(i's) featuresC_i. Then I will analyze and process the information forwarding data, and obtain the corresponding edge e of each blog post. Then I will use Edge set E and user set U to construct the graph G; The edge weights w are obtained by reproducing the independent cascade model to apply it to train our own data The designed tag is the ID set of the corresponding message sender U_s, and the tags of all users are extended to the same dimension to obtain multiple tags L for each user U_s, and each one-dimensional tag is marked as l. After dimension reduction, the coding is brought into the graph convolution network (GCN) and graph attention network (GAN) to obtain the training results and conduct a case analysis. After collecting the data, the independent cascade model is used to calculate the edge weights to obtain the features of graph structure , and then the user features and information features obtained from data processing are integrated to combine the three influencing factors (user features, information features and graph structure features) which are then imported into the GAT or GCN for training. The trained label is the reduced dimension label dimension obtained after the original label has been trained by the encoder/decoder self-coding layer. Then label with reduced dimension is restored to the predicted real tag with the same latitude as the original tag through the threshold value, and the evaluation parameters are calculated with the original tag to evaluate the model.

3.1 Dataset

According to the literature survey, the non-literal features of users include popularity, activeness, contacts, etc. This study used Zhang J and his team's public microblog data set Zhang, Liu, Tang, Chen, and Li (2013) in their research for filtering and research, which contains the non-literal features mentioned above and literal features such as the contents of the blog. This data set randomly selected 100 users from Weibo from September 28 to September 29, 2012 as seed users, and crawled their associated users in turn, collecting 1,776,950 users, 308,489,738 associated relationships, and 300,000 Weibo posters in total.

3.2 User feature processing

The dataset contains 14 user characteristics C, including 9 digital characteristics, These characteristics can represent a number of causal factors, for instance number of two-way followers could represent contacts, number of fans could represent popularity. Since the whole research process ignores the impact of time, the *created_at* characteristic (the time feature) is deleted. Because the serial number containing province represents the location of the user, therefore the location feature is deleted. Furthermore, the digital feature 1,0 replaces gender and verified with only two values, but unfortunately I do not collect data with people who do not identify themselves as males or females so I simply use two values, and the Pandas package in python is used to directly process the user set U which contains 10 features, as shown in Table 1.

Table 1: Characteristics and Instance

Characteristic	Instance	Numerical feature or not
id	1657151084	T
bi$_{f}ollowers_{c}ount$	0	T
city	5	T
verified	False	F
followers$_{c}ount$	33	T
location	Shanghai	F
province	31	T
friends$_{c}ount$	162	T
name	JACKJONES	F
gender	M	F
created$_{a}t$	2009-10-29-22:20:41	T
verified$_{t}ype$	-1	T
statuses$_{c}ount$	0	T
description	Weak	F

However, because the digital features of province and gender represent the state of the feature value but shows no evidence that Shanghai is larger than Beijing as 1 larger than 0 pure numbers have relation of size which may interfere with model judgment and learning. Therefore, these two features are replaced by one pot vector in the subsequent processing.

Because the feature above contains two text features, name and description, it cannot be directly substituted into calculation learning in training. This article uses the method of Bidirectional Encoder Representation from Transformers (Bert) to convert text into vectors.

Bert is a pre-trained text information processing model. It uses the pre-training task to enhance the model with mask to learn semantics, and introduces prediction of the next sentence to allow multiple sentences to be merged into one input. After that, the deep training layer uses powerful examples to support bidirectional transformer training to get word vectors that can express semantics well.

This paper refers to the pre-trained model Devlin, Chang, Lee, and Toutanova (2018) Bert Base, Chinese, to comprehensively train the above models to obtain the vectors of the above two features name and description.

Each message has its own corresponding content which contains long sentences. In order to substitute this influencing factor, the content of the message or blog into the whole message prediction process, this paper uses Jieba word segmentation to remove stop words to reduce the interference of invalid similar words on training, and then uses Bert for word vector conversion in the same way.

Then the information content messages are merged into the corresponding node features to obtain 784 dimensional text features. In order to reduce the computational complexity and balance the information content and other digital features of the node, this paper uses self-coding dimension reduction processing. An autoencoder self-coding model is trained with input as output samples, which is used in data denoising and dimension reduction. By choosing appropriate dimensions and sparse constraints, the self-encoder can learn more accurate coding results than PCA and other models. Moreover, unsupervised learning can be carried out from data samples without additional feature engineering. To achieve this goal of reducing the feature dimension, various parameters selected in this paper are shown in Table 2.

Table 2: Autoencoder hyper-parameter

Name of hyperparameter	Value of hyper-parameter
input$_{s}ize$	784
hidden$_{s}ize$	8
output$_{s}ize$	784
first$_{l}ayer_{a}ctivation$	ReLU
second$_{l}ayer_{a}ctivation$	sigmoid
optimizer	Adam
loss	MSE
epochs	5
batch$_{s}ize$	128

3.3 Edge feature processing

I filtered the collected data and clear all user IDs that do not exist in the user set, as shown in Figure1 As shown in the figure, every two lines of data correspond to one blog post, and the information content features have been integrated into the node features. The red box is U_s, and the corresponding blue box is U_r. It can be seen that each message may be forwarded by one user, there may be multiple messages, or there may be no messages, covering all possible situations in reality.

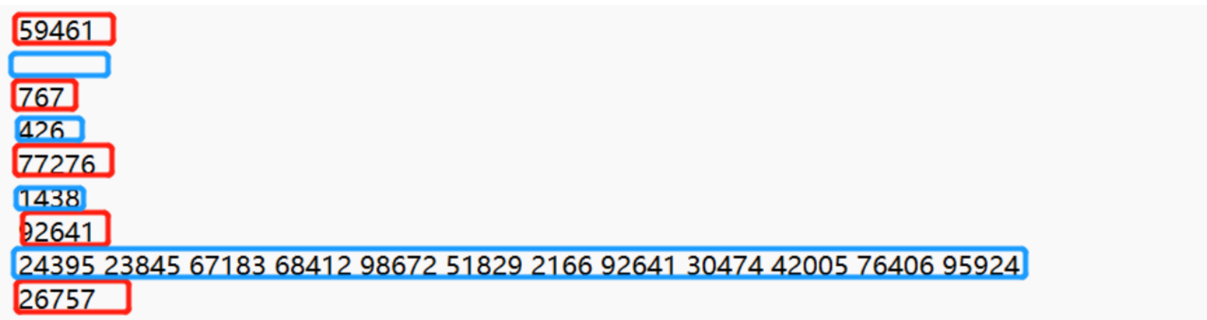

Figure 1: Edge data instance

In this paper, for each U_s and U_r, there is an edge existing in U_s and U_r, such as the fourth data point in Figure1, the existing edge $e_{92641,24295}$, $e_{92641,23845}$, $e_{92641,67183}$ and so on. Therefore, every blog post can be forwarded through the edge. Each edge forms a set of edges. Here, it needs to be clear that what this paper establishes is a forwarding blog graph, and the social network graph of followers can be established from the collected data, but when it is substituted into the model studied in this paper, the weight of the edge that does not generate forwarding behavior through the edge is 0, which has no impact on the results in the final graph neural network training. The two graphs are considered the same in the subsequent training, and are collectively referred to as the social network graph G.

3.4 Edge weight processing

We set parameters and apply them directly according to the independent cascade model Saito, Nakano, and Kimura (2008) (subsequently referred to as the IC model), and use the first 70% of the message forwarding data for model training. The initial value of each $k_{v,w}$ is set as a random value of 0.1-0.3 to obtain the result distribution shown in Figure 2.

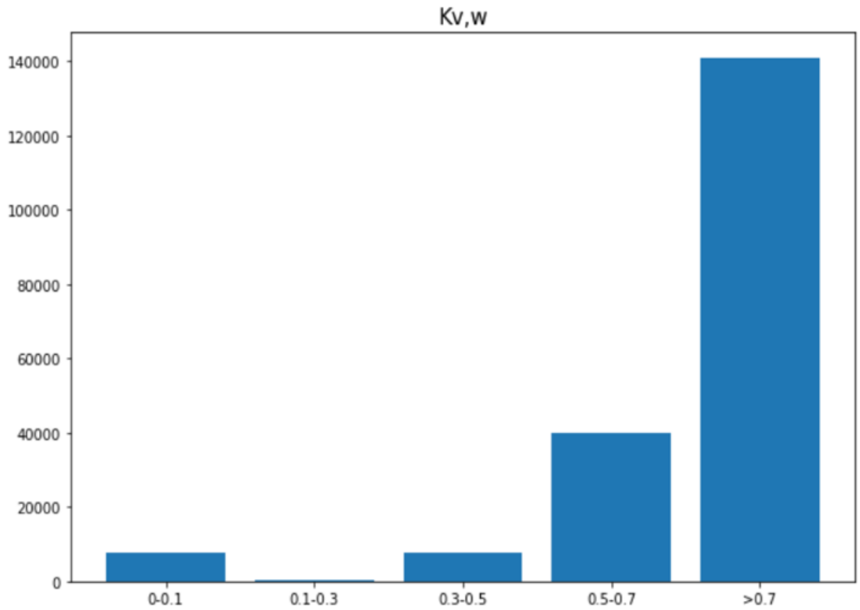

Figure 2: $k_{v,w}$ Distribution after IC model

As shown in the Figure3, the ordinate represents the number of edges and the abscissa represents the weight. Most of the weights in the edge are greater than 0.7, which means that the probability of message forwarding through this edge is high without considering other factors as the independent model assumes that probability only influenced by the edge between two nodes. There are also a small number of edges whose weights are less than 0.1, which means that there will be a small number of edges without considering other factors. The figure shows the statistical results. The first 1000 of more than 100,000 edges are selected as shown in Figure3. As shwon, the weight of each edge is calculated by the IC model iterative algorithm, and the value is distributed in a decentralized manner, independent of the edge serial number. Most of the values are more than 0.7, and a few values are distributed around 0.3, corresponding to the statistical results in Figure2.

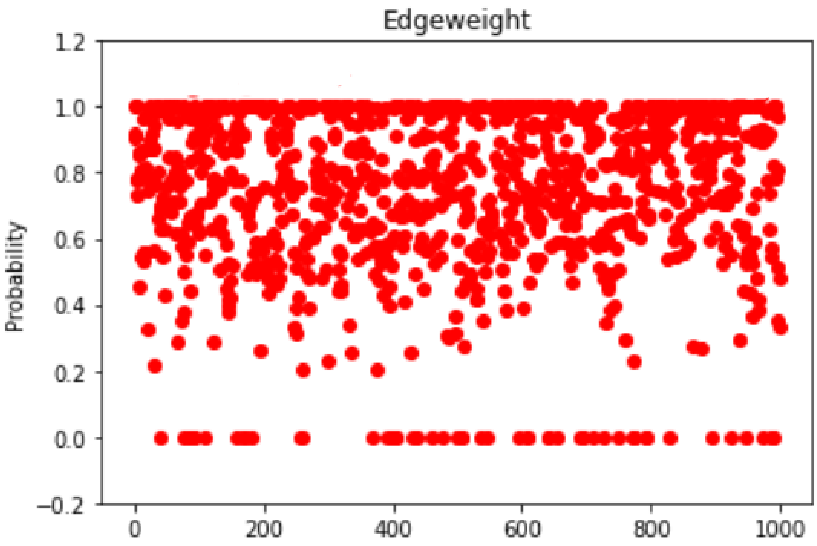

Figure 3: $k_{v,w}$ values of first 1000 edges

The iterative process of selecting an edge of the algorithm is observed to test the success of the algorithm, as shown in Figure 4. The iteration process of this edge is shown in the Figure4 , and the ordinate represents the probability value. It can be seen from the Figure4 that the P_w of the node w corresponding to this edge converges to 1.0 after about 15 iterations, and the weight $k_{v,w}$ corresponding to this edge converges to about 0.96 after around 40 iterations. It can be seen that both of them converge, conform to the results given in the original text, and the model is reproduced successfully. I used the data result $k_{v,w}$ with about 45 iterations as the final weight $k_{v,w}$ of each edge. So far, all data preprocessing work has been completed, and three factors - user characteristics, information characteristics and graph structure characteristics - have been integrated.

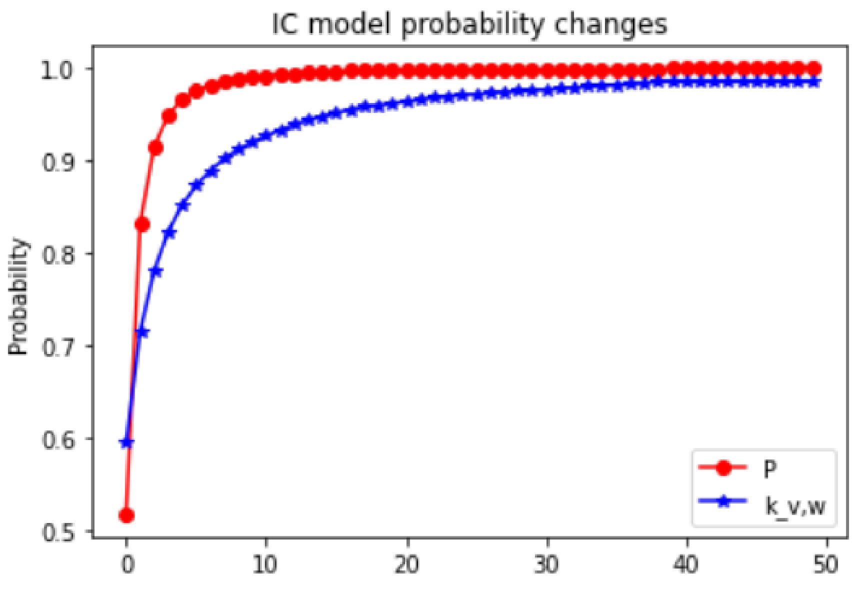

Figure 4: Iteration process of one edge

3.5 Multi-label design

In this paper, we need to predict whether the corresponding U_r of the U_s forwards the corresponding sent message. Therefore, the label of each U_r is designed as the union of all the IDs of the corresponding U_s . However, because the number of messages forwarded by each U_r is different, it is also possible that the user did not forward any messages, resulting in different dimensions of multiple labels, which cannot be substituted into the graph neural network for calculation. Therefore, we use the MultiLabelBinarizer in sklearn to convert all user tags into a 0-1 vector of the same dimension. The MultiLabelBinarizer can place all the labels appearing in the dataset as elements in each node label. If the label exists, the value of the element is set to 1, otherwise it is set to 0. Each user has 1579 dimensional labels. Because the graph of the dataset in this paper is sparse, each U_r 's dimension with a median value of 1 in 1579 is about 10 dimensions, which leads to lazy training, so that the model can output 0 on all dimensions with an accuracy of up to 99%. Therefore, this paper uses the encoder decoder model again.

The reduced dimension label can be obtained through the encoder net network. After saving, it is used as the predicted label in GNN, whose value is a floating point number. Then the trained model is saved. Finally, the predicted label is substituted into the decoder of the saved model and returned to the 1579 dimension label for model evaluation. Because the encoder decoder model cannot make the predicted value completely equal to the pre-prediction value, this paper sets a threshold value. A value higher than the threshold value in the prediction value is regarded as 1, and a value lower than the threshold value is regarded as 0, similar to the activation function principle. In this paper, the number of dimension is reduced to 3 dimensions.

3.6 ICGNN experimental parameters

After the IC model predicts the weight of the finished edge, this paper uses the graph after microblogging data set processing to carry out experiments. The ICGNN model is substituted to set the necessary parameters, as shown in Table3.

Table 3: ICGNN hyper-parameter

Name of hyperparameter	Value of hyper-parameter
Number of convolution layer	1
input$_{s}ize$	117
hidden$_{s}ize$	32
output$_{s}ize$	1
output$_{t}unnel$	3
Regularization method	L2
Regularization coefficient	0.5
loss	MSE
learning rate	0.008
epochs	60
threshold	0.05
Train set rate	0.7
optimizer	Adam

3.7 Experimental results

The experimental results obtained after training with the above parameters are shown in Figure5 and 6. As shown, the difference between the results of GAT and GCN is not significant. The loss decreases rapidly from the beginning of training, but GAT is stable at about 0.35 at the 30th epoch and GCN is stable at about 0.35 at the 45th epoch. For the recall rate . GAT experienced a peak and then decreased to 0.5135, and GCN slowly increased to 0.5285. The epochs used were similar to the above losses. The accuracy of GAT is 0.6935 and that of GCN is 0.6884.

It can be seen that there is no significant difference between the final effects of the two algorithms, but GCN requires more rounds, probably because it is a full batch training mechanism and the training parameters are less than GAT. However, GAT only uses the first order neighbor nodes for each training, so smooth training is not easy. This is also one of the possible reasons for the peak of GAT. But in a word, both algorithms are more suitable for the graph neural network part of this model.

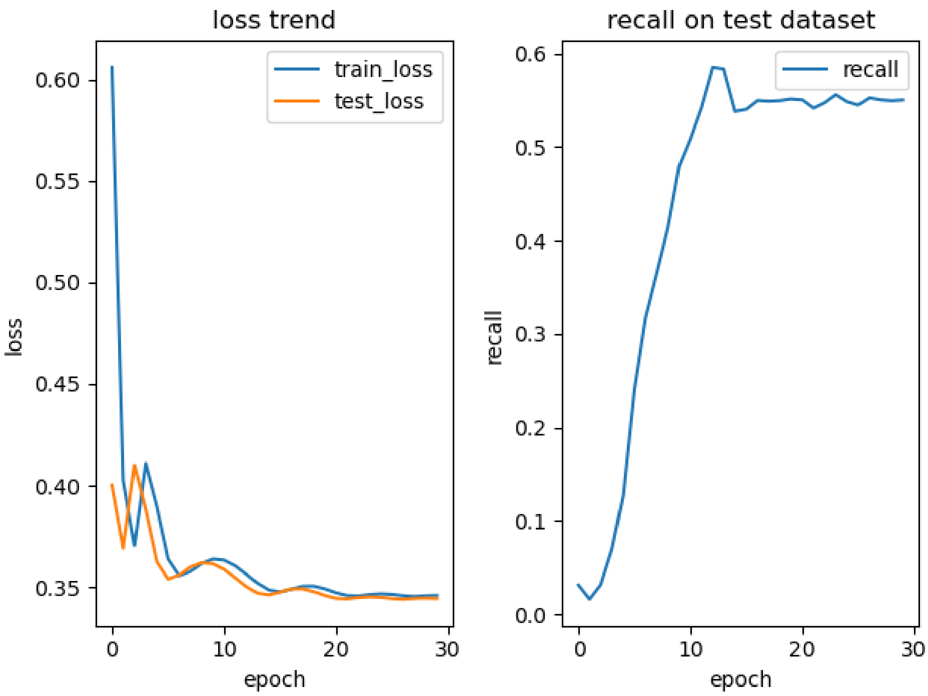

Figure 5: GAT results

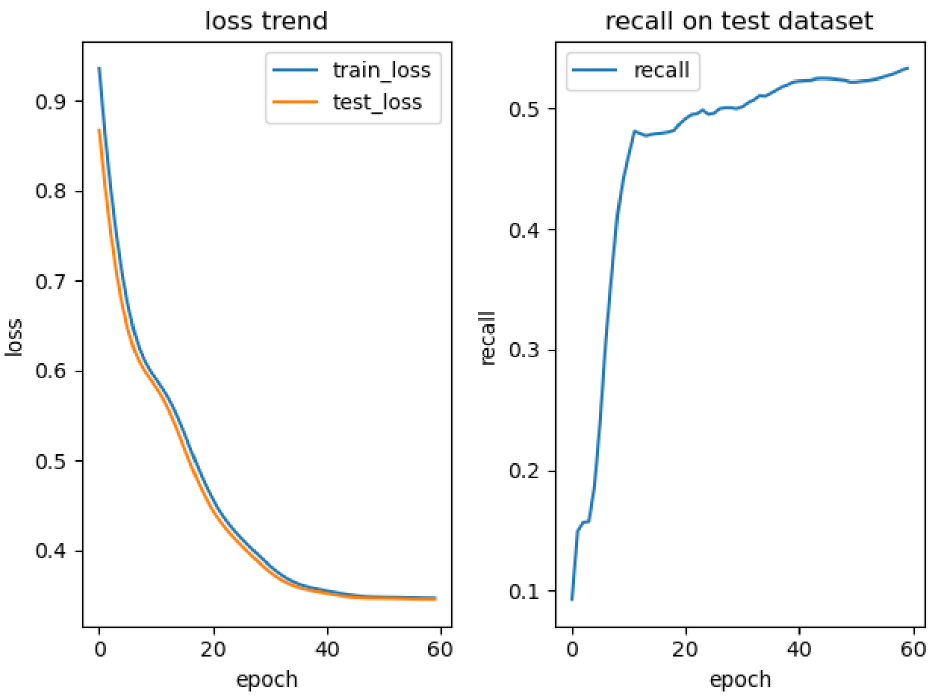

Figure 6: GCN results

3.8 Case study

I select a user with moderate blog posts in the test set for case analysis. This paper selects user with ID 82685 and forwards a total of 9 microblogs of U_s. The results of the prediction are shown in Figure 7, which shows 7 microblogs can be successfully predicted in 9 dimensions, respectively, with IDs of 73570, 16214, 82536, 52334, 73556, 73553, 37798, 17783, and 41501,except for 37798,17783.

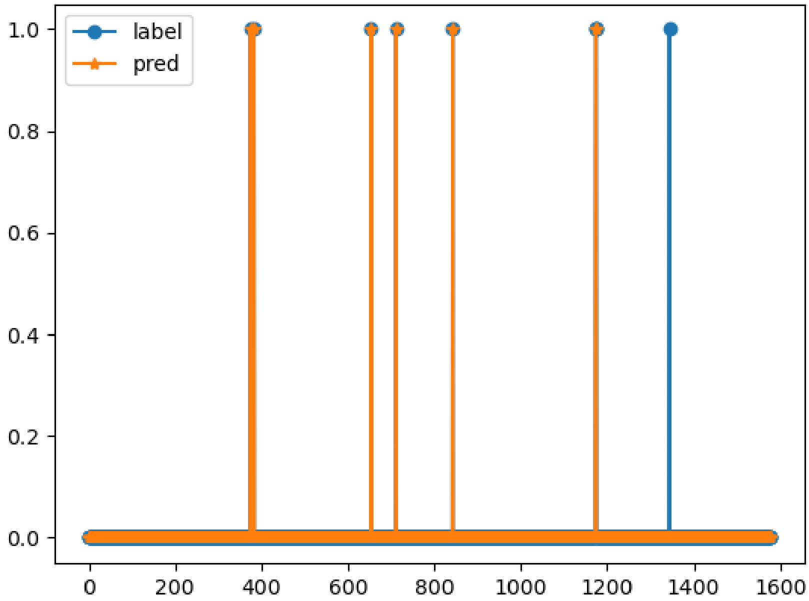

Figure 7: Case study results

The target user with ID 82685 has an interest in understanding life and a yearning for sunshine, and its name and characteristics have made important contributions to labeling the user and understanding the user. They do not have many fans, and the attention is maintained at a normal level. Although there is no authentication, it can be seen from the users' status that it is a normal non big V user, mainly browsing and forwarding blog posts.

The 7 users successfully predicted by this model have common characteristics in each important feature. Their attention is normal, they have millions of fans, and their status is very active, reflecting the characteristics of the time they spent online and high volumn of blog posts. Further their names all involve life topics, mainly female topics . Most of contents of the description include women, life and youth orientation topics. The textual features here include the user's personal details, and the content of the blog post, which are the results that can be easily displayed after manual refining. It is worth mentioning that the weights of the edges between 7 users and the 82685 are above 0.87, which shows that they also have a high probability in the aspect of the graph structure.

In addition, the remaining two users who failed to predict success, in the above features, aside from the fact that their attention remained normal, their number of fans was small, their *status* (feature) was not active enough, and they did not introduce themselves too much (few words in *name* and *description*, their content was not strongly related to women or life topics that 82685 was interested in, and because these two user characteristics were not strongly related to the users whose micro-blogs was forwarded by 82685 and neither were the corresponding blog features, it is understandable that we didn't predict success. In addition, the weight of these two users and the edge of 82685 is low, indicating that their forwarding probability is low. However, it is speculated that the reason why the 82685 actually forwards these two users that failed to be predicted is that they may have a close offline relationship, so 82685 forwarded their messages to support friends' blog posts. This factor cannot be predicted by our ICGNN model.

From the above data results, it can be seen that the calculation results of the opposite side weight of the IC model not only can facilitate the prediction of the subsequent GNN model, but also verify the applicability of the IC model. The two nodes with high weights are similar in various other features.

For all the users' features, gender, number of fans, active status, name and Chinese features, and edge weight have a greater impact on whether users forward. This argument can be concluded from the data results. But then it can be observed through experiments by eliminating non important features, re forecasting and comparing again, which is more scientific and persuasive. The authentication type should be used to verify whether the user is a zoombie user, such as the account that created to click farming without any subjective movement. However, it is speculated that all users are basically the same in these two features because the user tend not to complete personal profile, so the effect of these two features is low. The number of followers can be used as another factor to detect information diffusion. If the number of followers reaches thousands or even tens of thousands, it may be determined that they are zoombie users, which is also an area that can be studied later. This feature of friends does not have much essential relations in common. The two characteristics of gender and location do not have much in common neither, but this could be as the amount of data is insufficient. Here, it refers to the amount of data dedicated to these two characteristics, such as gender topics, or statistics that are only forwarded by men. In this case, the user 82,685 is male but interested in female life topics, which cannot summarize conclusions from examples since males can interested in males topic or females topic, which can be a direction for future research.

In a word, the ICGNN model is in line with the actual situation in this case. The above features have a positive impact on the results, and the impact of the features is analyzed to further verify the applicability and accuracy of this model. Further, the success prediction rate of this case was 77.8%, and it can be analyzed that it has low probability that the two failed cases are resluted of the model. In addition, this case study can provide guidance for future research.

4 Conclusion

The main task of this paper is to predict the rule of information dissemination in social networks. In general, this paper does the literature review, the collection and processing of public microblog data sets, the processing of non-data text features using existing mature natural language processing models, the reduction of dimensions using existing mature self-coding models for subsequent processing, the design of classified multi tags, the replication of existing papers for their own use, and the design of a complete set of models to predict the law of information dissemination, through which a representative case study was made.

From the experimental results, the accuracy of the whole model is 69%, and the recall rate is 52%, which shows that the whole model has a good prediction effect and is suitable for the prediction of the information transmission law of the social network. In the case study, descriptive statistics were used to analyze the influence of each factor. The results show that text features and edge weights had a greater impact on the results, that certification had a smaller impact on the results, and gender had a greater impact on the results. From the results, it is concluded that the paper reappears successfully, and has a greater positive impact on the final results.

From the actual experimental results, this paper proposes a new prediction model that integrates edge features and node features. In the same model, user data features, blog features and graph structure features are integrated, which has value in practical applications.

Social networking applications have reached deep into people's lives, and the field of deep learning is also booming. Although the model proposed in this paper can effectively predict the rules of information transmission of microblogging data sets, there are some problems that can be further studied in the future. In this paper, only one data set is applied, so multiple data sets could be applied to verify the robustness of the model, but result from only one data set still does not affect the practicality of the model in this paper. Based on the analysis of the case study, in the future researches can further delete some factors that have less impact on the results, and conduct further experiments to obtain the degree of impact of factors scientifically and accurately, and focus on specific factors for targeted data collection and verification. Furthermore, for the influencing factors, more factors such as age can be collected, and the influencing factors can also be graded to explore the causal relationship between various factors, so as to enrich the whole model. To improve the accuracy of the model, the threshold can be designed and calculated mathematically, so that there is an accurate threshold for each data set, making the probability of correctly predicting propagation more accurate. Social networks have become a key everyday source of information . Further graph neural network has developed rapidly in the field of deep learning, and achieved good results in various practical applications.The research in this paper hopes to promote the combination of graph neural network and social networks, so as to improve the effectiveness of message supervision and the ability to predict rules.

References

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ding, Y., Yan, S., Zhang, Y., Dai, W., & Dong, L. (2016). Predicting the attributes of social network users using a graph-based machine learning method. *Computer Communications*, *73*, 3–11.

Guille, A., Hacid, H., Favre, C., & Zighed, D. A. (2013). Information diffusion in online social networks: A survey. *ACM Sigmod Record*, *42*(2), 17–28.

Hu, H., Mao, H., Hu, X., Hu, F., Sun, X., Jing, Z., & Duan, Y. (2015). Information dissemination of public health emergency on social networks and intelligent computation. *Computational intelligence and neuroscience*, *2015*.

Jiao, C. (2019). *Research on false information detection method for microblog information propagation time and structure characteristics* (Unpublished doctoral dissertation). Xinan Jiaotong University.

Jiuxin, C., Jianglin, W., Wei, S., Bo, L., Xiao, Z., & Junzhou, L. (2014). Information diffusion analysis and prediction on micro-blog. , *37*(4), 779–790.

Liu, Y., Bao, Z., Zhang, Z., Tang, D., & Xiong, F. (2020). Information cascades prediction with attention neural network. *Human-centric Computing and Information Sciences*, *10*(1), 1–16.

Mashhadi, A. J., Mokhtar, S. B., & Capra, L. (2009). Habit: Leveraging human mobility and social network for efficient content dissemination in delay tolerant networks. In *2009 ieee international symposium on a world of wireless, mobile and multimedia networks & workshops* (pp. 1–6).

Michalski, R., Kazienko, P., & Król, D. (2012). Predicting social network measures using machine learning approach. In *2012 ieee/acm international conference on advances in social networks analysis and mining* (pp. 1056–1059).

Murphy, C., Laurence, E., & Allard, A. (2020). Deep learning of stochastic contagion dynamics on complex networks.

Saito, K., Nakano, R., & Kimura, M. (2008). Prediction of information diffusion probabilities for independent cascade model. In *International conference on knowledge-based and intelligent information and engineering systems* (pp. 67–75).

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Philip, S. Y. (2020). A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, *32*(1), 4–24.

Yang, C., Sun, M., Liu, H., Han, S., Liu, Z., & Luan, H. (2018). Neural diffusion model for microscopic cascade prediction. *arXiv preprint arXiv:1812.08933*.

Yangfan, G. (2021). Analysis of research object of false information management. *China's water transport: the second half of the month*.

Zhang, J., Liu, B., Tang, J., Chen, T., & Li, J. (2013). Social influence locality for modeling retweeting behaviors. In *Ijcai* (Vol. 13, pp. 2761–2767).