Yelp Fraud Detection and Explanation Based on Graph Neural Networks Algorithms

Muhao Chen
A20456889
mchen69@hawk.iit.edu

Abstract: Fraud reviews are common in the social platform, and many scholars have tried different advanced Graph Neural Networks (GNN) to detect these fraud reviews. Different from the simple Graph Convolutional Networks (GCN) and Graph Attention Networks (GAT), this paper used CARE-GNN Model[1] proposed by Dou and Liu(2020), due to the advantages of reinforcement learning and multi-relation threshold selection. And finally the CARE-GNN Model have reached 69% accuracy and 74% AUC. Additionally, this paper apply tree-based model, XGBoost to explain how the features explain the result of predictions.

1. Introduction

As is acknowledged, reviews fraud is very common on the Internet platform, including yelp and Amazon. A large number of merchants create hundreds of accounts/robots to write comments. However, limited by the technical methods, these fake and useless reviews were difficult to detect initially.

However, with the development of technology and big data, most reviews fraud are detected. The main methods include exploring the relationship between graph nodes, generating more effective features and advanced sampling techniques. In this report, I use yelp data-set[1] and improve the accuracy through relations processing and additional features.

2. Literature Review

Different from Euclid data, the composition of fraudulent reviews is non-Euclidean data, which requires GNN[2] instead of traditional machine learning and deep learning models. In recent years, many scholars have begun to study different structures of GNN. Due to the complexity of data relationships and models, more and more optimized models have been proposed.

2.1. Conventional GNN Model

The most common GNN model includes the GCN and GAT. Kipf and T. N. (2016) proposed GCN model[3], which was widely used in the fraud detection field. And Veličković (2017) added an attention module to change the weights between nodes, which is named GAT[4].

Traditional GNN model, including GCN and GAT, have a significant advantages of sampling and aggregation. However, in the actual application process, scientists will also encounter problems such as unbalanced samples, high time complexity, and multi-relational connections.

2.2. Modern GNN Model

Because of the various requirements in the actual research, many scholars proposed some optimal models. For example, Liu (2021) proposed a Pick and Choose Graph Neural Network (PC-GNN for short)[5] for imbalanced supervised learning on graphs, which solved the lacking in yelp fraud labeled data. And Dou and Liu (2020) proposed an enhancing graph neural network-based fraud detectors[1], which performed more excellently than GCN and GAT.

Obviously, in this paper, yelp data have some complex problems and I use the enhancing graph neural network-based fraud detectors to solve these problems.

3. Data Description

3.1. Data Source

I get the data from dgl.data.fraud, which includes hotel and restaurant reviews filtered (spam) and recommended (legitimate) by Yelp, which has been widely used in some scholars' paper.

3.2. Data Overview

The reviews are regarded as nodes in the graph, and there are three styles of edges, including R-U-R, R-T-R and R-S-R. The detailed information shows in Table 1.

Tabela 1. Information of Data-set

Name	Explanation	Number
Nodes	number of reviews	45,954
R-U-R	reviews posted by the same user	98,630
R-T-R	reviews under the same product with the same star rating	1,147,232
R-S-R	reviews under the same product posted in the same month	6,805,486
pos-label	number of positive labels	6,677
neg-label	number of negative labels	39,277
feature-num	number of features	32

4. Features Generation

Actually, the generated features are provided in the yelp data-set, which were 32 handcrafted features from Rayana (2015)'s working[6]. Rayana generated some new effective features of users, products and reviews based on the metadata provided, part of which are showed in Label 2.

Tabela 2. Part of Features Generated

Name	Correlation(H/L)	Details	Style
MNR	H	Max. number of reviews written in a day	User
ERD	L	Entropy of rating distribution of user's	User
PR	H	Ratio of positive reviews (4-5 star)	Product
WRD	H	Weighted rating deviation	Product
EXT	H	Extremity of rating	Reviews
PCW	H	Percentage of ALL-capitals words	Reviews

Features are important to the result of prediction. And in the graph, the features of reviews themselves are not enough to reflect the actual situations. If we add the other two objects, user and product, more features are generated easily, which are more useful. For example, if one user have posted a lot of reviews in a short time, the probability of the fake reviews is improving.

5. Basic GNN Model

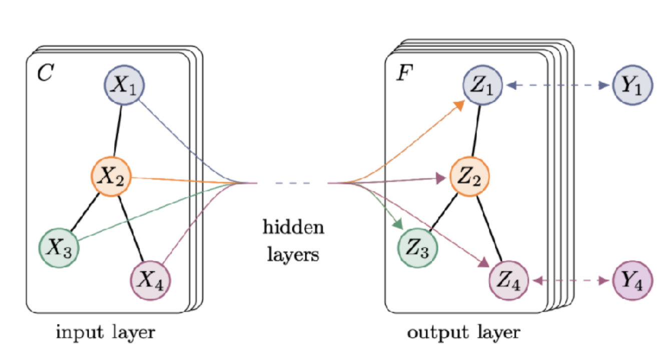
5.1. GCN Model

GCN is a network structure, which transform the initial features of nodes into new features. Because, the features of neighbors of nodes can influence the features of nodes. The aggregation formula is:

$$f(H^{(l)}, A) = \sigma\left(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{1}$$

The aggregation occurs in every hidden layer, and some sub-graphs are generated through sampling. Different sub-graphs hold different value of features, and these difference could increasing while the hidden layers are increasing. Therefore, GCN could simulate the spread and influence between nodes. Show in Figure 1.

Figura 1. GCN Model Structure



5.2. GAT Model

GAT was defined due to the different weights of edges. Actually, GAT is just a different aggregation function with attention over features of neighbors, instead of a simple mean aggregation. The weights calculating formula is:

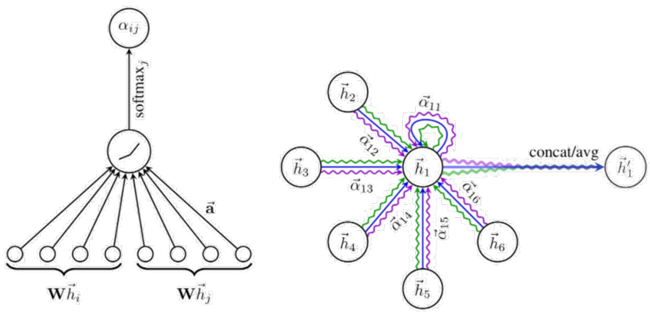
$$e_{ij}^{(l)} = \text{LeakyReLU}(\vec{a}^{(l)^T}(z_i^{(l)}|z_j^{(l)})) \tag{2}$$

And we should normalize the weights of different edges, therefore, the normalization formula is:

$$\alpha_{ij}^{(l)} = \frac{\exp(e_{ij}^{(l)})}{\sum_{k\in\mathcal{N}(i)}\exp(e_{ik}^{(l)})} \tag{3}$$

The neighbors of nodes will be calculated the similarity between them to get the value of significance. However, the attention mechanism would take a long time to calculating, due to the large number of edges. Furthermore, there is a LeakyReLU function help to increase the importance and decrease the useless edges' influence. Show in Figure 2.

Figura 2. GAT Weights Calculating

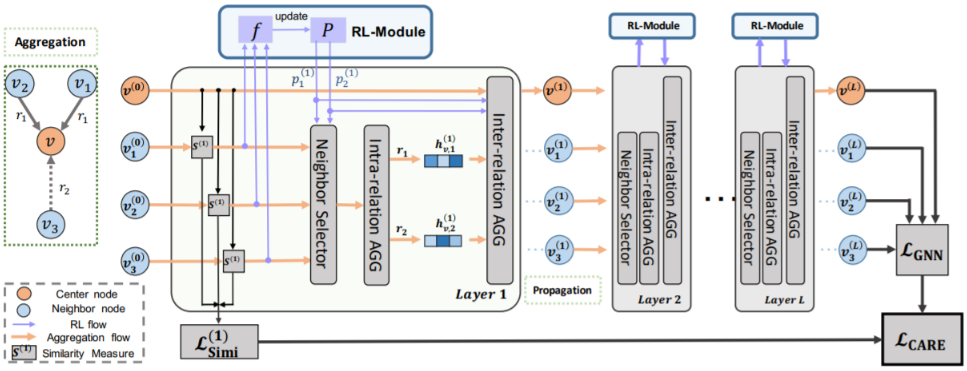


6. CARE-GNN Model

6.1. Overview

Dou and Liu (2020) proposed an enhancing graph neural network-based fraud detectors[1], which performed more excellently than GCN and GAT. CARE-GNN Model add two useful mechanisms, three relations' neighbor selection and reinforcement learning. Show in Figure 3.

Figura 3. CARE-GNN Model Structure



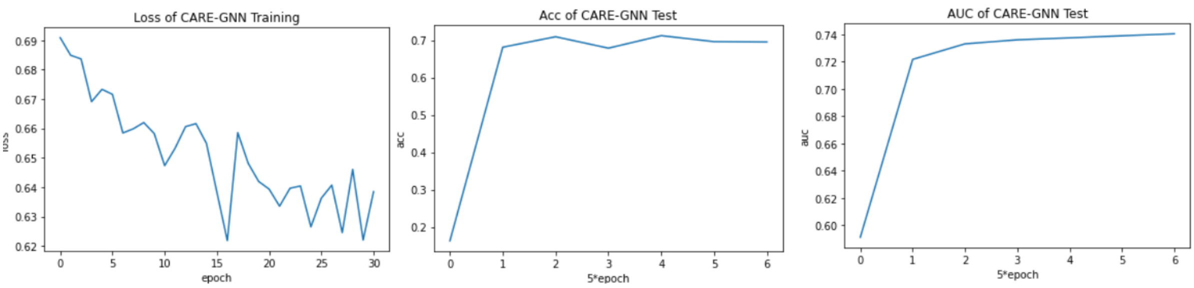
Actually, there are three types of relations between reviews, which are referred above. And three types of relations need the different attention mechanism, which precisely reflects the actual influence between nodes.

Furthermore, CARE-GNN Model adds reinforcement learning, which will give a positive or negative feedback after each epoch. And if the feedback is positive, the selection modular will keep the parameters, otherwise dropping.

6.2. *Train and Evaluation*

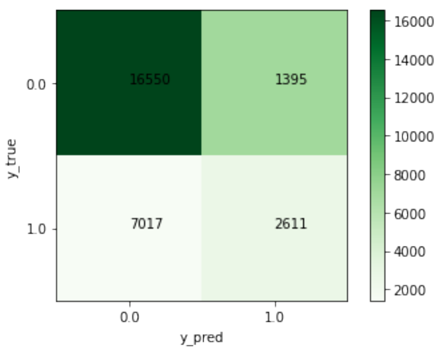
Before training, we performed data pre-processing and set some Hyper-parameters. We normalized the features and under-sampled the normal reviews. Besides, we designed the three layers and set the 31 epochs to train the model. The loss is decreasing while the accuracy and AUC are increasing. The final accuracy is 69% and AUC is 74%. Show in Figure 4.

Figura 4. Information While Training



After training, we put the predicted data and labeled data into the evaluation of confusion matrix. As we can see, approximately 65% fraud reviews are detected, however, there are also some normal reviews are predicted to be the fake ones. Show in Figure 5.

Figura 5. Confusion Matrix



7. Explanation of Predictions

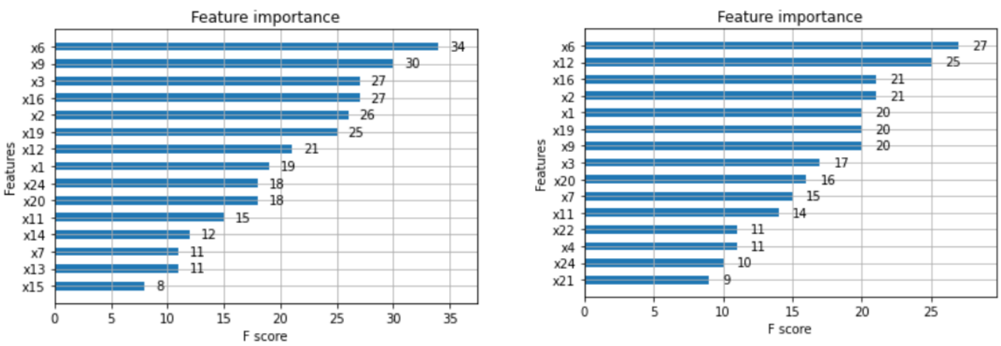
7.1. XGBoost

XGBoost is a tree-based model, which selected the features to split based on the gain of information. Different from the simple classification tree, XGBoost supported L1 and L2 regularization. Besides, XGBoost is a boosting algorithm, which uses Taylor formula to study the residual of previous tree, which is more accurate.

7.2. Features Ranking

I establish two XGBoost models to help me figure out 1.which features could explain the prediction? 2. which features could explain the errors of prediction? Because, most of time, we should consider why the GCN and CARE-GNN predict these nodes to the specific labels. And we should avoid the errors of prediction, while using the features, which could easily cause the misunderstanding. Show in Figure 8.

Figura 6. Explain the prediction(Left) Explain the Errors(Right)



Obviously, x6 and x16 are both ranked high, which means these two features could explain the prediction excellently and would not easily cause the errors. And x3 are ranked high in explaining prediction, but ranked low in explaining errors, which means this feature could cause the errors easily.

Referências

1. Dou, Y., Liu, Z., Sun, L., Deng, Y., Peng, H., Yu, P. S. (2020, October). Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In Proceedings of the 29th ACM International Conference on Information and Knowledge Management (pp. 315-324).
2. Akoglu, L., Tong, H., Koutra, D. (2015). Graph based anomaly detection and description: a survey. Data mining and knowledge discovery, 29(3), 626-688.
3. Kipf, T. N., Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.
4. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y. (2017). Graph attention networks. arXiv preprint arXiv:1710.10903.
5. Liu, Y., Ao, X., Qin, Z., Chi, J., Feng, J., Yang, H., He, Q. (2021, April). Pick and Choose: A GNN-based Imbalanced Learning Approach for Fraud Detection. In Proceedings of the Web Conference 2021 (pp. 3168-3177).
6. Rayana, S., Akoglu, L. (2015, August). Collective opinion spam detection: Bridging review networks and metadata. In Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining (pp. 985-994).