A Dynamic Default Prediction Framework for Networked-guarantee Loans

Dawei Cheng[1], Yiyi Zhang[1], Fangzhou Yang[2], Yi Tu[1], Zhibin Niu[3] and Liqing Zhang[1*]
[1]Shanghai Jiao Tong University, Shanghai, China
[2]Seek Data Inc., Shanghai, China [3]Tianjin University, Tianjin, China
{dawei.cheng, yi95yi, tuyi1991}@sjtu.edu.cn, zniu@tju.edu.cn, zhang-lq@cs.sjtu.edu.cn

ABSTRACT

Commercial banks normally require Small and Medium Enterprises (SMEs) to provide their warranties when applying for a loan. If the borrower defaults, the guarantor is obligated to repay its loan. Such a guarantee system is designed to reduce delinquent risks, but may introduce a new dimension risk if more and more SMEs involve and subsequently form complex temporal networks. Monitoring the financial status of SMEs in these networks, and preventing or reducing systematic loan risk, is an area of great concern for both the regulatory commission and the banks. To allow possible actions to be taken in advance, this paper studies the problem of predicting repayment delinquency in the networked-guarantee loans. We propose a dynamic default prediction framework (DDPF), which preserves temporal network structures and loan behavior sequences in an end-to-end model. In particular, we design a gated recursive and attention mechanism to integrate both the loan behavior and network information. Then, we uncover risky warrant patterns by the learned weights, which effectively accelerate risk evaluation process. Finally, we conduct extensive experiments in a real-world loan risk control system to evaluate its performance, the results demonstrate the effectiveness of our proposed approach compared with state-of-the-art baselines.

CCS CONCEPTS

• **Applied computing** → **Economics**; • **Information systems** → **Data mining**.

KEYWORDS

Default prediction, Graph mining, Networked-guarantee loans

ACM Reference Format:
Dawei Cheng[1], Yiyi Zhang[1], Fangzhou Yang[2], Yi Tu[1], Zhibin Niu[3] and Liqing Zhang[1]. 2019. A Dynamic Default Prediction Framework for Networked-guarantee Loans. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM'19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3357384.3357804

*Corresponding author.

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11...$15.00
https://doi.org/10.1145/3357384.3357804

1 INTRODUCTION

For banks and financial administrations, it is becoming more and more important to predict and prevent the systematic risks of networked guarantee loans. the loans' existing credit criteria are aimed primarily at independent key players that lag behind the demand of small and medium-sized enterprises (SMEs)[19]. In order to meet the loan criteria of the issuer, groups of SMEs warrant each other to enhance their financial security. While, in practice, they may provide inaccurate or manipulated data, and subsequently cause interrelated risk factors. As more and more companies are involved, they form complex directed-network structures [26]. Thousands of guaranteed credit networks of varying complexity have long co-existed and evolved over time. This requires an adaptive strategy to prevent, identify and eliminate systematic crises [31].

Traditional credit default prediction methods are mainly based on financial probability models[5, 7], e.g., logistic regression and credit scorecards, which employ a shallow network for the basic profiles of the loan. With the advances of the most recent deep-learning technologies developed, novel and deep algorithms emerge, e.g., [12, 15, 35], to improve the accuracy of the prediction by integrating behavioral information or increasing model capacity

However, these traditional techniques have two key shortcomings: 1) They extract features mainly from the historical profile and behavior of individual SMEs, so the likelihood of network-linked credit outages can not be determined. 2) The models construct static features from temporary credit records by time sliding windows and apply feature representation methods to learn features in each window. This means that they manually break a credit default prediction system into two separate phases: feature-representative learning and classifying, makes them highly ineffective and computationally intensive to achieve a productive response in real-word applications.

For this reason, we propose a dynamic end-to-end framework to effectively predict borrowers' defaults by mining the temporal records of networked-guarantee loans. The intuition behind this technique is that a borrower's probability of default in guarantee networks is also influenced by other nodes, based on our observations and feedback from financial experts. This is because the guarantor takes on the debt obligation if it's warranted borrower fails to repay the loans. So if the guarantee can not repay the loans, the loan lender can enforce its guarantors. on this occasion, the delinquency can spread like a virus throughout the network. The contagion increases both the possibility of the occurrence of risks and the transfer of guarantor risks. In particular, in times of economic downturn, some companies will face operational difficulties

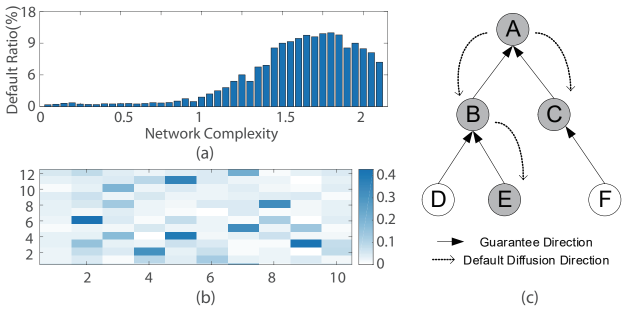

Figure 1: The statistical results of the loan dataset: (a) the default ratio of guaranteed loans of network complexity from 0 to 2; (b) the default ratio by random sampled 120 sub-network; (c) empirical study of default diffusions.

and the financial crisis will create a domino effect: the outage phenomenon can quickly spread across the network, which can cause many enterprises to get into an unfavorable situation.

By aggregating extensive warranty and credit history records across the network, we can therefore investigate the likelihood of a delinquency event by examining the different patterns of defaults. Fortunately, the complete loan record data provided by our cooperate institutions offers us a unique opportunity to tackle the problem. With the access to the large-scale and high-quality loan records, we first conduct a set of experiments to validate the feasibility of our assumption, i.e., a company's loan default is affected by other firms in the guarantee network. We first conduct statistical analyze on the default status of a company who involved in different situation of guarantee networks, and then we empirically analyze company repayment in a typical network with domain experts.

Figure 1a visualizes the statistical results, where the delinquency rate increases in along with the complexity of networks. It is observed that the default probability is highly diversified according to different sub-networks, as shown in Figure 1b. Moreover, in over six month's empirical analysis, the default diffusion along with edges in guarantee network is also demonstrated. As shown in Figure 1c, company A (a paper producers) failed to repay a loan of 0.7 million dollars in Dec 2015, and subsequently caused it guarantor (company B and C, lumber and packaging manufacturer respectively) defaults after four months. Finally, the risk diffused through company B into E (a paper distributor) within six months. As a result, this set of data analysis validates our original assumption.

However, to implement such a default prediction, there are still many challenges: 1) data outdated, the financial information reserved in the loan can't keep pace with the rapid change of company's business operation. 2) dynamic default prediction model, developing an effective recognition model is not trivial, as it is difficult to represent patterns of temporal networks; 3) knowledge discovery, with the massive guaranteed loan records, knowledge discovered from prediction models are import in financial literature, while this is a common limitation of most deep learning models.

In this paper, we design, implement and deploy a dynamic default prediction system based on data mining results from massive guaranteed loan records. The system consists of three main modules: 1) data pre-processing, which generates loan behavior features and temporal guarantee networks; 2) default prediction, which studies from temporal network to model the risky guaranteed loan patterns, combines with loan behavior features in one model and infers the possibility of the delinquency loan events; 3) risky pattern discovery, which visually analyze the risky patterns and explores latent business implications from learned weights.

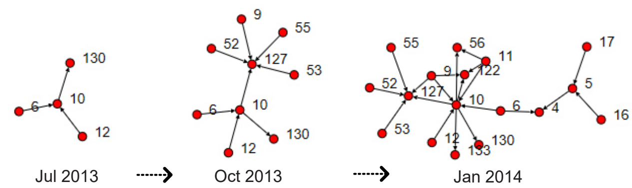

Figure 2: Visualization of temporal guarantee networks.

The main contributions of the paper are summarized as follows:

- We provide the first attempt on predicting bank guarantee loan defaults by adding objectively temporal network structure using an end-to-end learning framework.
- We design and implement the Dynamic Default Prediction Framework (DDPF), which enables the model to represent temporal networks structures. We also propose new attention and RNN based deep architecture and demonstrate that the learned representation can improve the performance of default prediction.
- We thoroughly evaluate DDPF in over six months' empirical study. The research was evaluated in a real-world loan risk management system. The result demonstrates that DDPF can effectively predict defaults and discover risky patterns.

In the rest of the paper, we describe the preliminaries and preprocessing in section 2 and 3, detail the prediction model in section 4, show experiments in section 5 and 6. Section 7 and 8 summarize the related work and conclusion.

2 OVERVIEW

2.1 Preliminaries

Definition 2.1. ***Guarantee Network.*** A guarantee network GN is a directed graph $G = (V, E)$, where $V = \{v_1, v_2, \cdots, v_m\}$ is a set of SMEs, and $E = \{e_1, e_2, \cdots, e_n\}$ is a set of guarantee relationships (edges).

For each $e_i \in E$, it associates with three properties: 1) amount, which indicates the quota of warranty; 2) start time, which is the effective date of the contract; and 3) end time, which indicates its expiry date.

Definition 2.2. ***Temporal Guarantee Network.*** A temporal guarantee network TGN can be defined as a time-ordered graph sequence $TG = \{G_1 \rightarrow G_2 \rightarrow \cdots \rightarrow G_n\}$, where $G_i = (V, E, t_i), 1 \leqslant i \leqslant t$, is a guarantee status with involved SMEs V, relationships E, and timestamp t_i.

Figure 2 gives an example of the concepts. The nodes are SMEs, and the edges with arrows indicates warrant issued from guarantors to borrows. The figure illustrates the network over time from Jul 2013 to Jan 2014. The dashed arrows indicate their sampled sequence.

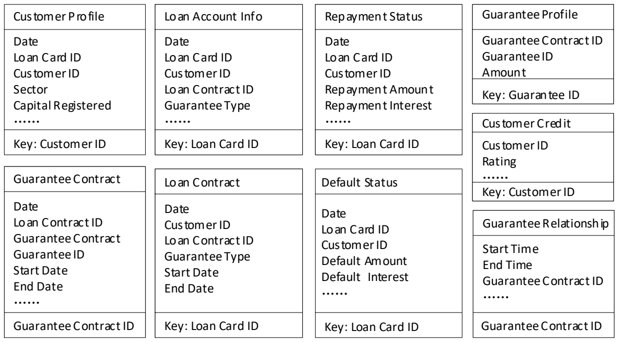

Figure 3: The dictionary of loan datasets. It includes nine tables from relational databases, which generally covers loan records, guarantee status, user basic profile and default events.

Definition 2.3. ***Loan Default(Delinquency) Event.*** A loan default event in this paper refers to a SME's delinquency of repayments over due.

2.2 Problem Definition

We now formalize our default predicting problem as follows:

Given a set of loan and repayment records, a temporal guarantee network TG, and time period t_i & t_{i+1}, for every SMEs, we want to infer the possibility of loan default events, based on the loan and guarantee status from t_1 to t_i. The objective is to achieve a high accuracy of the default prediction, as well as explore the risk patterns of guaranteed loans.

3 DATA PREPROCESSING AND ANALYSIS

We collect loan records over a four-year period from a major financial institution. The identity and personal data of the clients are encrypted and replaced with unique client IDs. In the dataset, the basic profile of SMEs can be acquired, such as the company scale, loan information like the guarantee ID and loan credit. In this section, we first introduce the data and preprocessing, and then analyze the temporal property of networked guarantee loans.

3.1 Preprocessing and Feature Extraction

Figure 3 gives brief illustration of raw loan records. It contains nine tables from the data management system, which generally falls into four categories: SME's profile, loan records, guarantee status and default event history. We highlight the key of tables by unique ID names to illustrate the relationship among these records.

Afterwards, in data preprocess phase, we construct 1) *loan behavior sequence* $Q = \{Q_1, Q_2, \cdots, Q_n\}$, 2) *temporal guarantee networks* $TG = \{G_1, G_2, \cdots, G_n\}$ and 3) *user features*, by aggregating over the nine tables. Particularly, loan behavior feature includes: current time window, default status, loan type, loan amount, historical loan times, historical loan amounts, etc. They are calculated on the basis of all loan records created before the current time window. Temporary guarantee networks are generated from the start time to the end time of the warranty contract. Each record specifies a network edge, which affects between the start and the end time. User features include SME type, register capital, number of employees, company scale, etc. Normally companies are required to update their basic information when making loan applications, therefore we utilize the latest user data as user features.

It should be noted that the base profile may not be fully trusted as the SMEs may provide the loan with outdated or even inaccurate information. However, the guarantee network is trustworthy because we can build it from the recording system. Therefore, the following section, we continue to statistically analyze the consequences of the guarantee network.

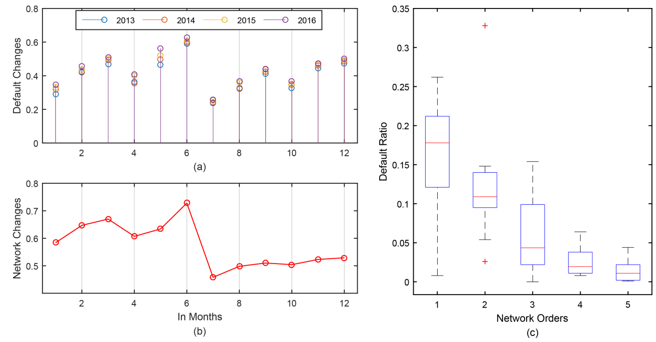

Figure 4: Empirical analysis results of the loan dataset. (a) the number of new default events in each month from 2013 to 2017; (b) the average number of new edges in each month; (3) the default ratio of delinquent SMEs' n−th(1-5) order neighborhoods. Noted: we scale the value of y-axis for better illustration.

3.2 Data Analysis

We perform an empirical analysis on the loan dataset with networked guarantee loan records, from 01/01/2013 to 31/12/2016. We divide all the loan records by a given time window size, which is set to 1 month by default. We construct features in each time window according to the method mentioned in Section 3.1, and find the following observations on temporal and structural properties of default events.

Temporal Properties. Figure 4a shows the loan default ratio status according to the time. X-coordinate indicates Jan to Dec, y-axis is the number of new default events from 2013 to 2016, while the average number of new edges (guarantee relationships) in network is presented in Figure 4b. It is clear that the number of new default events shows annual periodicity, and reaches peak on the last month of each quarter. This phenomenon also appears in Figure 4b, shows the synchronicity of guarantee status and default events.

Structural Properties. We collect all SMEs who fail to repay loans in each time window, then present the average default ratio of the delinquent SMEs' n−th order neighborhoods in the next time window, as shown in Figure 4c. We can see that the far away from the defaulted SMEs, the lower the average default ratio of the firm is. This follows our intuition that the default probability of a firm is also influenced by its adjacent nodes in temporal guarantee networks.

4 LOAN DEFAULT PREDICTION

Given a set of loan and repayment records, temporal guarantee network, the prediction system infer the possibility of default events. Based on the intuition and empirical study results in section 1, a SME's default event is not only depend on financial status of itself, the situation of guarantee and other companies from the temporal network also play an important role. Therefore, we propose a Dynamic Default Prediction Framework (DDPF) for networked-guarantee loans. In this section, we present the model architecture firstly, which are detail described in Figure 5. Then we present input and temporal network embedding, attention and prediction network, respectively.

4.1 Input and Embedding Layer

The input of DDPF $x = \{x_1, x_2, \cdots, x_n\}$ contains loan behavior sequence $Q = \{Q_1, Q_2, \cdots, Q_n\}$ and temporal guarantee networks $TG = \{G_1, G_2, \cdots, G_n\}$, where Q_i, G_i indicate the loan behavior and guarantee network in i^{th} time window.

As described in section 3.1, loan behavior sequence normally contains loan features: such as amount, interest, frequency, etc., which can be summarized as current (active) and historical parts. So, q_i is described as a pair $< active_i, historical_i >$, where $active_i$ indicate the status in current time window, contains: repay amount, interest, default status, etc. $historical_i$ describes the accumulate features of a user loan behavior, such as total loan amount, loan times and default times.

G_i represents network features of SME in current time window i, it includes one-hot network structure vector and position feature p_i. Position stores the number of default nodes in the nearest 1 to j^{th} order neighborhoods. For example, if defaulted nodes in a company's 1^{st} order adjacent is denoted as $p_{i,1}$, and j^{th} order adjacent is $p_{i,j}$, then we reach the position feature as $p_i = \{p_{i,1}, p_{i,2}, \cdots, p_{i,j}\}$.

Different features are modeled separately in our proposed DDPF, and then are concatenated together in order to preserve the overall behavior and network representations. Each input is represented by a multi-hot vector $\{x_i^1, x_i^2, \cdots, x_i^F\}$ which represents loan behavior (loan type, loan amount, default status in current and historical parts) and network features (one-hot network and position vector). F is the dimension of features of each sequence, and x_i^f is the f^{th} feature. The feature can be a one-hot, multi-hot or numerical feature. For input sequence i, embedding layer transforms the vector $x_i = [Q_i, G_i]$ into a low-dimensional dense vector $res_i = [q_i, g_i]$ by linear mapping (q_i and g_i represent the embeddings of Q_i and G_i respectively), as illustrated in Equation 1:

$$res_i = \left[W_{emd}^1 x_i^1, W_{emd}^2 x_i^2, \cdots, W_{emd}^F x_i^F\right], W_{emd}^f \in \mathbb{R}^{d_{emb}^f \times V_f} \tag{1}$$

where d_{emb}^f is the embedded vector dimension of the f^{th} feature and V_f is the size of the vocabulary. The dimensionality of the network representations are significantly reduced by the network embedding layer into a much smaller size, thereby regulating the model.

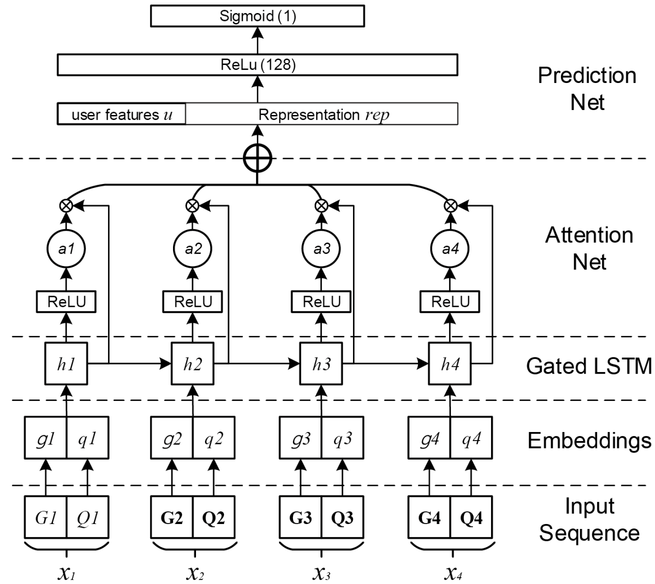

Figure 5: The model architecture of DDPF. It takes loan behavior sequence, user features and temporal guarantee network as input, infer the probability of loan delinquency as output, contains embedding layer, Gated LSTM, attention and prediction net.

4.2 Recursive and Attention Net

Given the sequential characteristics of user behavior in bank loans, the output of the network embedding layer is fed into a recursive neural network (LSTM) layer. In LSTM, a hidden state h_t is updated by the current entry and the previous hidden state h_{t-1} in a recursive formula. Considering the characteristics of two parts (q_t and g_t) of an input sequence (res_t), follow Yabo's work [27], we introduce Gated LSTM into DDPF, in which the loan behavior q_t and network embeddings g_t are treated differently. In particular, q_t can not specify a specialty of the network, but it does reflect the importance of each loan behavior. Therefore, it can be considered as an active signal in the recursive model (described in equation 2, 3 and 5). That is, q_t has an extensive effect on the element extraction, memorization and forwarding. It indicates the characteristics of the network, is the only input of LSTM (described in equation 4). The fully gated recursive model can be formulated as below:

$$i_t = \sigma(W_{gi} g_t + W_{qi} q_t + W_{hi} h_{t-1} + b_i) \tag{2}$$
$$f_t = \sigma(W_{gf} g_t + W_{qf} q_t + W_{hf} h_{t-1} + b_f) \tag{3}$$
$$c_t = f_t \cdot ct - 1 + i_t \cdot \tanh(W_{gc} g_t + W_{hc} f_{t-1} + b_c) \tag{4}$$
$$o_t = \sigma(W_{go} g_t + W_{qo} q_t + W_{ho} h_{t-1} + b_o) \tag{5}$$
$$h_t = o_t \cdot \tanh(c_t) \tag{6}$$

where i_t, f_t and o_t denote the input, forget and output gates of the t^{th} object respectively. c_t represents the activation vector of the cell.

The output of recursive model is another array of vectors $h = \{h_1, h_2, \cdots, h_N\}$. Over the recursive model, we employ an attention mechanism. We consider each vector h_i as the i^{th} node (SME) in networks, and represent the learned vector as an additive weighting of the low-dimensional vector of all the nodes. The learned weights of attention mechanism allow the node to allocate proper lending according to its importance in the current time. Mathematically, it has the following form:

$$\begin{aligned} rep_s &= \sum_{t=1}^{N} a_t h_t \\ a_t &= \frac{\exp(ReLU((h_t, u, q_t : \omega))}{\sum_{t=1}^{T} \exp(ReLU(h_t, u, q_t : \omega))} \end{aligned} \tag{7}$$

where a_t denotes the learned hidden neural weight h_t, $ReLU(\cdot;\omega)$ represents a fully connected neural network layer of attention mechanism. The input of $ReLU(\cdot;\omega)$ includes: profile features u, the t^{th} hidden state h_t, and the loan behavior embedding q_t. rep_s denotes the vector of output sequences and we concat the learned rep_s and u (features of SME profiles) into unified representation rep, which is a 128 dimensional vector.

4.3 Prediction Layer and Optimization

This paper aims to predict loan default events. Based on the above unified representation rep, we then apply them into the prediction layer, as shown in Figure 5. We define the loss function as described in Equation 8:

$$\begin{aligned} \mathcal{L}(\alpha) = -\frac{1}{N}\sum_{i=1}^{N} [& y_i \log(\text{predict}(rep_i, u_i : \alpha)) \\ &+ (1 - y_i)\log(1 - \text{predict}(rep_i, u_i : \alpha))] \end{aligned} \tag{8}$$

where rep_i is the output of the neural layer of attention mechanism, which denotes the representation of i^{th} node (SME), u_i represents the user features, and y_i represents the ground-truth of the i^{th} node, which is marked as 1 if the loan defaults and 0 otherwise. predict(rep_i, u_i) is a prediction function that maps rep_i and u_i to a continuous numerical value between 0 and 1, which is the predicted default probability of the SME in current time window. In the implementation, we employ predict$(, : \alpha)$ as a shallow neural network (two layer ReLU and one layer sigmoid).

Our proposed DDPF can be optimized by the standard SGD process. In our implementation of learning process, we apply the Adam optimizer [21] to update the parameters. The initial learning rate is set to 0.0001, and the batch size to 512 by default.

5 EXPERIMENTS

In this section, we conduct extensive experiments to evaluate the effectiveness and efficiency of our proposed methods. We first describe the experimental settings. Then, we give experiment results of default prediction compared with other baselines, which is the main task of this paper. After that, the effects of dynamic models, attention layer are tested respectively. Finally, we provide the effectiveness evaluation of end-to-end model.

5.1 Experimental Settings

Datasets: We collect data from a major financial institution in East Asia, during 01/01/2013 and 31/12/2016. It includes 112872 nodes(SMEs), with 124957 edges (guarantee relationships). We observed that most of loans are repayed in month. Hence, we aggregate the behavior feature with one month time window and mark the delinquency loans as target label in month.

Compared Methods and Parameter Settings: The following state-of-the-art methods are used to highlight the effectiveness of DDPF on the loan dataset, including Wide [24], Wide and Deep [11], CNN-max [36], crDNN [32], INDDP [9]. Our model has several variations, includes: DDPF-deepwalk/node2vec/DHPE [16, 29, 37] (in which replace the temporal guarantee network with static network of the first time window), DDPF-static and the proposed model DDPF-all. All parameters are setting as their default recommendations, which are detailed in supplemental materials.

Evaluation Metrics: The performance of the proposed prediction model is evaluated by AUC (Area Under the Curve), Precision@k and MAP (Mean Average Precision).

Precision@k is used to evaluate the performance of default prediction, which means the prediction precision of top k nodes. The formula of Precision@k is:

$$Precision@k = \frac{|\{i | i \in \mathbf{V}_p \cap \mathbf{V}_o\}|}{\mathbf{V}_p} \tag{9}$$

where $\mathbf{V}_p$ is the set of predicted top k defaulted nodes, $\mathbf{V}_o$ is the set of observed default nodes and $|\cdot|$ represents the size of the set.

Mean Average Precision(MAP) is used to evaluate the performance of default diffusion, which measure the rank accuracy of predicted node list. The formula of MAP@k is:

$$\begin{aligned} AP@k(i) &= \frac{\sum_{j=1}^{k} Precision@j(i) \cdot \delta_i(j)}{\sum_{j=1}^{k} \delta_i(j)} \\ MAP@k &= \frac{\sum_{v_i \in V} AP@k(i)}{|V|} \end{aligned} \tag{10}$$

where $Precision@j(i)$ is the Precision@j for node v_i, and the $\delta_i(j)$ indicates the v_j is diffused by delinquent node v_i.

In the KS test we tried all possible threshold probabilities from 0 to 1 with the step size 0.01. To determine the most effective threshold, we validate the prediction result with the target labels in retrospective testing. An F1 score is calculated to reflect the effectiveness of various thresholds.

5.2 Comparison Results

We present the performance comparison between proposed DDPF and baseline methods for the default prediction task in this section. The 2012 records are used as training data and then we predict the defaults over the next three years of the month. We indicate the average AUC in Table 1.

The first five lines of Table 1 contain the results of some current baselines. CNN-max and INDDP models are proving to be competitive across all baselines. Lines 6-8,13 presents the AUC of our proposed DDPF and some of its sub-models. The performance of DDPF-nolb is close to INDDP. DDPF-lblstm and DDPF-lbatt perform much better than the state-of-the-art baselines and DDPF-nolb. The behavioral properties prove to be effective when used in both the gates of recursive and the neural network of attention mechanism. DDFP-all surpasses all other methods.

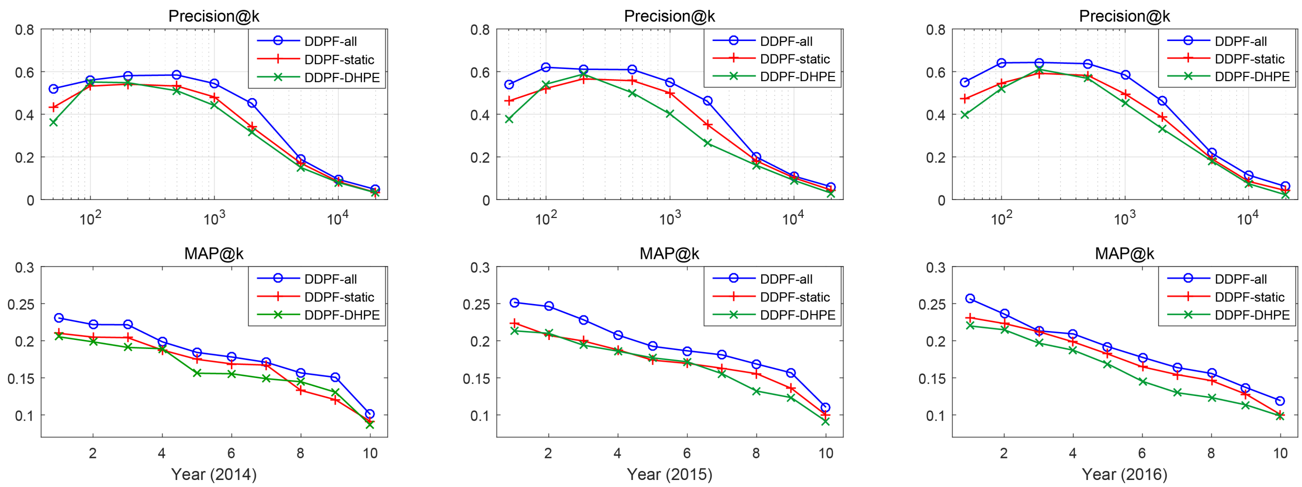

Figure 6: Precision@k results on default prediction and MAP@k on default diffusion, respectively. We compare with static model (DDPF-static) and embeddings pre-trained (DDPF-DHPE) approaches. The results show our proposed dynamic end-to-end (DDPF-all) method achieves the best performance in all time windows.

Table 1: Comparison the default prediction results of Different Models.

	AUC(2014)	AUC(2015)	AUC(2016)
Wide	0.75509	0.77751	0.78195
Wide & Deep	0.76464	0.79825	0.81053
CNN-max	0.77645	0.80049	0.81492
crDNN	0.77429	0.79565	0.81054
INDDP	0.77942	0.80205	0.81267
DDPF-nolb	0.77455	0.79372	0.81205
DDPF-lblstm	0.78155	0.80015	0.82031
DDPF-lbatt	0.80501	0.80004	0.82151
DDPF-deepwalk	0.78848	0.80624	0.81341
DDPF-node2vec	0.80140	0.81048	0.81659
DDPF-DHPE	0.81940	0.82184	0.82245
DDPF-static	0.80263	0.81686	0.82087
DDPF-all	**0.82097***	**0.83742***	**0.84135***

Lines 9 through 11 report the results of DDPF with pre-trained network embeddings. DDPF-DHPE outperforms deepwalk and node2vec, demonstrates the effectiveness of dynamic model of network embeddings. The same phenomenon is proved by the last two rows, results of DDPF-all is significantly better than those of DDPF-static. In addition, compared with pre-trained embedding based methods, the result improvement of DDPF-all is especially significant and the end-to-end learning can help with the model performance improvement is strongly demonstrated. The pre-trained network embedding mainly depends on the simultaneous occurrence of the nodes, so that only the similarity of the nodes can be determined by the pre-trained model. On the contrary, DDPF-all can extract more explicit information from nodes such as delinquent diffusion over edges.

5.3 Effects of Dynamic Models

For a temporal network, its representation can be learned by either static network in each time window or pre-trained by dynamic network embedding methods. By embedding temporal network in a recurrent neural network, our proposed model is able to extract more distinct information in an end-to-end model. We verify the above benefits by comparing pre-trained embeddings and static network models.

We then perform experiments on the benchmark dataset with DDPF-DHPE and DDPF-static methods, which are state-of-the-art pre-trained embeddings and static network models. The upper part of Figure 6 shows the precision@k of default prediction with different k. The end-to-end dynamic model outperforms other two baselines significantly. The experimental results demonstrate the effectiveness of proposed default prediction model.

In default diffusion experiments, we first select delinquent nodes in current time window, then predict the default probability of its guarantors in the next time window. The MAP@k results is shown in second parts of Figure 6. DDPF-static and DDPF-DHPE are shown to be competitive. DDPF-all performs much better than two baselines. This indicates that our methods can effectively capture the default diffusion information.

5.4 Performance Evaluation

In Figure 7a, the F1-score results are represented with the corresponding precision P and recall R in relation to various thresholds: 1) If the threshold is close to zero, all the test data are marked as default nodes. Thus, the precision is close to 0 and recall is near 1; 2) As the threshold increases, more and more instances are marked as positive, and we identify that threshold equal to 0.36 as F1 score reaches the maximum of 0.71. As a result, we use 0.36 as threshold in our system.

Then, we compare the efficiency of DDPF-all and DDPF-DHPE. In each time step the time costs are counted in comparison to

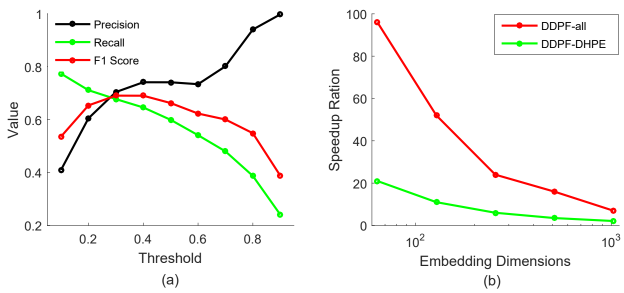

Figure 7: a) Results of KS-test on threshold selection, we chose 0.36 in our system. b) Efficiency comparison of DDPF-all and DDPF-DHPE with baselines (retrained by static models), shows advantages of utilizing dynamic models.

the base values (retraining by DDPF-Static). Figure 7b shows the average acceleration ratio with respects to different embedding dimensions of the network. We can see that the speedup effect is significant, DDPF-all can achieve more than 50 times acceleration ratio if the dimension is less than 100, compared with retraining by DDPF-Static. As the size of embedding dimension increases, the acceleration ratio decreases.

6 CASE STUDIES

Our proposed framework is deployed and evaluated in a real-world loan management system for early warning risky loans and pattern discovery. In this section, we present visualization results of embeddings and attention weights learned by the proposed end-to-end neural network. By tracking gradual changes of attentional weights, we try to discover the risk loan mechanism in the networked guarantee loans.

6.1 Network Layouts

Creating meaningful visualizations that layout a network in a two-dimensional space is one of the most important application of network embedding approaches. In this experiment, we divide nodes into three types of categories: 1) High risk, indicates the node has been defaulted in current time window or previous windows; 2) Medium risk, loan repayments of the SME is normal, but at least one of its neighbors delinquent its loans; 3) Low risk, not only the SME repay loans normally, but also it first order neighbors also repay normally. Laying out this guarantee network is very challenging as the three categories are very close together. We map them first into a low-dimensional space with DHPE and DDPF-all, then map the low-dimensional vector of vertices to a two-dimensional space with the t-SNE package [22]. Figure 8 shows the visualization with different embedding methods during the observation time window. The visualization with DHPE is not very meaningful because the nodes with the same rick level are not grouped together. This is because DHPE uses generalized SVD (GSVD) to ensure the high-order proximity, which does not use loan delinquent information in an end-to-end approach. DDPF-all performs quite well and generates meaningful layout of the guarantee network, nodes with same colors (risky levels) are distributed more tightly.

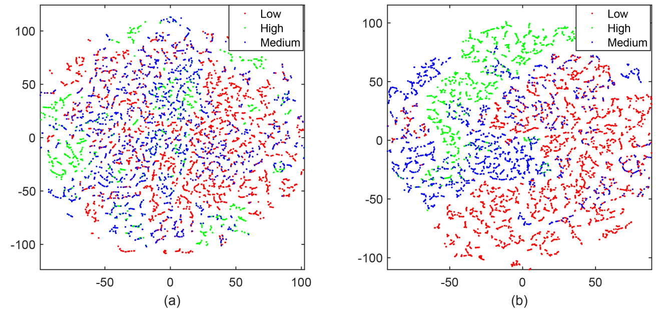

Figure 8: Visualization of the learned embeddings by t-SNE. DDPF-all performs better than DHPE, as nodes with same colors (risky levels) are distributed more tightly.

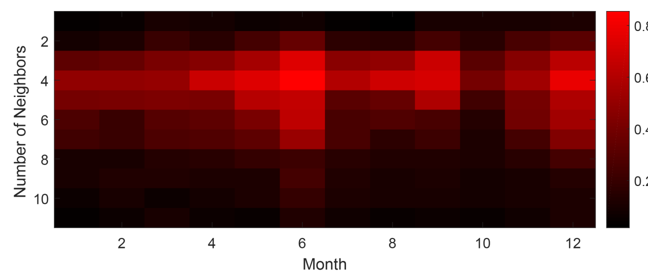

Figure 9: Case study results of the attention network. The node with around 4 adjacent nodes in every 3 and 6 month is generally highlighted.

6.2 Visualize Attention Weigths

In DDPF, we employ attention oriented pooling to determine the significance of different nodes in temporal guarantee networks within a given time window. We first randomly select nodes with a different number of neighbors to evaluate the effects of attention based pooling. The learned neural weights of attention mechanism from each time windows (months) with different number of neighbors are visualized in Figure 9. As we can see, 1) in column aspect, the attention layer highlights the weights of the nodes at the end of each quarter (especially around June and December), which is also reported in Figure 4; 2) in row aspect, nodes with different number of neighbors affect differently in attention mechanism. The node with 4 adjacent nodes is commonly significant because heat map are lighter than other types. This phenomenon is also reported in [25]. The above observations demonstrate that the attention layer in DDPF can successfully extract information regarding the risk patterns while eliminating irrelevant nodes.

6.3 Discover Risky Patterns

The highlighted attention weights can help us to study and focus on risk patterns more efficiently. It can be shown in the following three aspects: (1) Detect automatic motifs from high attention nodes. In particular, we use the gtrieScanner approach [30] for motif detection of about 4 nodes, since nodes with about 4 neighbors in Figure 9 are highlighted. (2) Match the motifs to the entire network and calculating the ratio for default companies. (3) Arrange the

Table 2: Statistical information for the high default motifs.

Motif ID	#19	#15	#20	#16	#17	#8	#3	#14	#5	#11	#2	#9
Motifs	1	4	1	4	4	74	169	6	151	22	312	24
Firms	4	10	4	28	18	165	238	24	304	138	410	176
default firms	4	9	3	18	11	79	110	9	89	32	95	26
Ratio for default firm	100	90	75	64	61	48	46	38	29	23	23	15
Ratio for default amount	100	100	100	55	75	56	53	47	58	49	64	44
Total loan amount	36	78	64	955	218	3259	5442	263	6975	4930	8919	3433
Total default amount	36	78	64	522	163	1829	2897	123	4072	2405	5686	1507

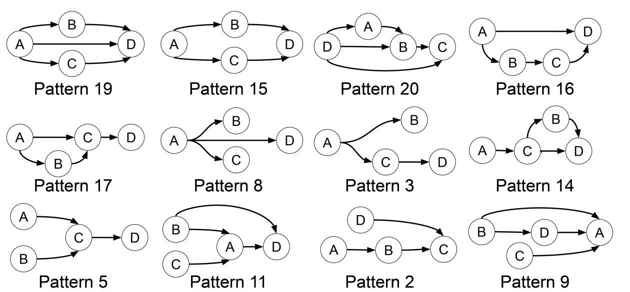

Figure 10: Risky 4-vertex-motif structures detected from a guarantee network (GN32). In which, patterns 15, 16, and 17 show single-input, single-output, and feed-forward structures.

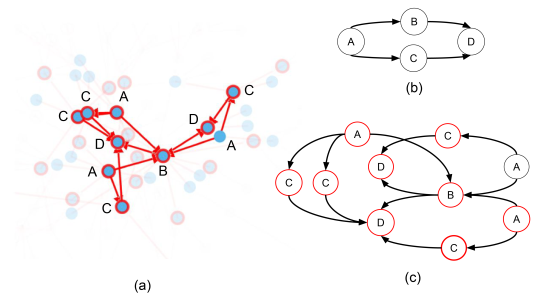

Figure 11: a) Pattern 15 motifs highlighted on the loan guarantee network. b) Pattern 15 model. c) Alternative way to discover pattern 15.

motifs in descending default order and mark top n as candidates of high default patterns. It would be time consuming to match all those motifs across the whole network. Theoretically, there are 199 possible combinations for 4-vertex moifs and and 9364 possible combinations for 5-vertex motifs in a directed network. Therefore, the learned neural weights of attention mechanism enable us to start from the high risky 4-vertex motifs.

We further demonstrate some representative cases to show the high default patterns selected by attention-based pooling, which are observed in deployment period. We select a typical subgraph coded as GN32, which consists of 103 companies; 36% of them take the 85% default amount out of the loan. Figure 10 indicates the risky pattern candidates identified from GN32, and Table 2 presents the statistical information.

As we can see, there are about 200 types of 4-vertex node motifs in total, but only 12 of them are detected as candidates with high risky patterns. Therefore, we could analyze only on the 12 motifs, instead of the over 200 motifs. In the empirical study phase, we find that some of these motifs are well-known by loan risk management experts. A group of others can be seemed as variations of smaller patterns. Taking motif 5 as an example, it is a combination of a joint liability guarantee loan with a single guarantee. Three of the motifs, 15, 16, and 17, come into focus for several reasons: (1) high delinquent rates for the patterns (between from 61% and 90% in the ratio for default company and 55% to 100% in the ratio for default amount); (2) a relatively small number of instances (4 or 5) are detected by the guarantee network; (3) the top five risk motifs show single input, single output, feed forward structures.

Figure 11 lists all the detected nodes of motif 15 in the entire network, where some of the nodes are collapsed. These high risky candidates are interesting; for example, pattern 15 appears five times in a sub-graph of guarantee networks, all companies failed to repay loans, who are involved in such guarantee structures (see Table 2). There is a high likelihood that it may be a loan fraud pattern exists several times, but the loan issuer fail to identify such a fraud pattern in advance.

7 RELATED WORK

We summarize the related works in two main areas: 1) credit risk evaluation, 2) network and financial risk.

Credit risk evaluation. Consumer credit risk assessment is often technically addressed in a data-driven manner and has been studied extensively [4, 13]. Since the seminal work "Partial Credit" model [23], numerous statistical approaches are introduced for credit scoring, such as [17, 18, 27]. More recently, [3] presents an in-depth analysis how to interpret the learned knowledge embedded in the neural networks using explanatory rules, and discussed how to visualize these rules. The authors in [20] combine debt-to-income ratios with consumer banking transactions, and use a linear regression model with time-windowed data to predict the delinquent rates in the near future. They claim a default prediction accuracy of 85% and can save cost between 6% and 25%.

Network and financial risk. Financial crises and systemic risk have always been a major concern for financial companies and governments, as extensive work has been explored [6, 14]. Networks or graphs represented by interconnected nodes and links between them are a good representation of modern financial systems, as they also have complex dependencies and relationships inside [1, 34]. The relationship between network structure and financial system risk is carefully examined and lessons learned: the network structure has little impact on the well-being of the system, but plays an important role in determining systemic risk and welfare in short-term debt [2, 9, 10, 28]. Although initial efforts have been made using network theory to understand fundamental problems in financial systems [8, 33], there is little work apart from the preparatory work on the risk assessment in the guarantee loan network [26, 28].

8 CONCLUSION

In this paper, we study an important problem of delinquent prediction for networked-guarantee loans. We propose a dynamic default prediction framework, which integrating temporal guarantee network, loan behavior sequences and SME features in LSTM and attention mechanism. Benefiting from better representation utilizing temporal networks and loan sequences, the learned embeddings and attentions could effectively accelerate the process of risky pattern discovery. We demonstrate it by case studies in GN32 and present the insights by empirical analysis on typical risky loan patterns. We conduct extensive experiments to evaluate the effectiveness of each sub-module and compare the proposed method with state-of-the-art baselines, the result verify the effectiveness of DDPF in default prediction and risky pattern discovery. To our best knowledge, this is the first work to address the guarantee network credit risk evaluation problem by considering both temporal network structure and loan behavior sequences.

ACKNOWLEDGMENTS

The work was supported by the National Basic Research Program of China (2015CB856004), the Key Basic Research Program of Shanghai Science and Technology Commission, China (15JC1400103, 16JC1402800), and China Postdoctoral Science Foundation.

REFERENCES

[1] Franklin Allen and Ana Babus. 2008. Networks in finance. (2008).
[2] Franklin Allen, Ana Babus, and Elena Carletti. 2010. *Financial connections and systemic risk.* Technical Report. National Bureau of Economic Research.
[3] Bart Baesens, Rudy Setiono, Christophe Mues, and Jan Vanthienen. 2003. Using neural network rule extraction and decision tables for credit-risk evaluation. *Management science* 49, 3 (2003), 312–329.
[4] Bart Baesens, Tony Van Gestel, Stijn Viaene, Maria Stepanova, Johan Suykens, and Jan Vanthienen. 2003. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the operational research society* 54, 6 (2003), 627–635.
[5] Cristián Bravo, Lyn C Thomas, and Richard Weber. 2015. Improving credit scoring by differentiating defaulter behaviour. *Journal of the Operational Research Society* 66, 5 (2015), 771–781.
[6] Markus K Brunnermeier and Martin Oehmke. 2012. *Bubbles, financial crises, and systemic risk.* Technical Report. National Bureau of Economic Research.
[7] Dean Caire. 2004. Building credit scorecards for small business lending in developing markets. *Bannock Consulting, http://www. microfinance. com/English/Papers/Scoring_SMEs_Hybrid. pdf* (2004).
[8] William K Carroll, Colin Carson, M Fennema, E Heemskerk, JP Sapinski, et al. 2010. *The Making of a Transnational Capitalist Class: Corporate power in the twenty-first century.* Zed books.
[9] Dawei Cheng, Zhibin Niu, Yi Tu, and Liqing Zhang. 2018. Prediction Defaults for Networked-guarantee Loans. In *2018 24th International Conference on Pattern Recognition (ICPR).* IEEE, 361–366.
[10] Da-Wei Cheng, Yi Tu, Zhen-Wei Ma, Zhi-Bin Niu, and Li-Qing Zhang. 2019. BHONEM: Binary High-Order Network Embedding Methods for Networked-Guarantee Loans. *Journal of Computer Science and Technology* 34, 3 (2019), 657–669.
[11] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems.* ACM, 7–10.
[12] Elena Dumitrescu, Sullivan Hue, Christophe Hurlin, and Sessi Tokpavi. 2018. Machine Learning for Credit Scoring: Improving Logistic Regression with Non Linear Decision Tree Effects. (2018).
[13] Eva Maria Falkner and Martin RW Hiebl. 2015. Risk management in SMEs: a systematic review of available evidence. *The Journal of Risk Finance* 16, 2 (2015), 122–144.
[14] Gerhard H Fischer and Ivo W Molenaar. 2012. *Rasch models: Foundations, recent developments, and applications.* Springer Science & Business Media.
[15] Trevor Fitzpatrick and Christophe Mues. 2016. An empirical comparison of classification algorithms for mortgage default prediction: evidence from a distressed mortgage market. *European Journal of Operational Research* 249, 2 (2016), 427–439.
[16] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 855–864.
[17] WE Henley and David J Hand. 1996. A k-nearest-neighbour classifier for assessing consumer credit risk. *The Statistician* (1996), 77–95.
[18] Cheng-Lung Huang, Mu-Chen Chen, and Chieh-Jen Wang. 2007. Credit scoring with a data mining approach based on support vector machines. *Expert systems with applications* 33, 4 (2007), 847–856.
[19] Ming Jian and Ming Xu. 2012. Determinants of the guarantee circles: The case of Chinese listed firms. *Pacific-Basin Finance Journal* 20, 1 (2012), 78–100.
[20] Amir E Khandani, Adlar J Kim, and Andrew W Lo. 2010. Consumer credit-risk models via machine-learning algorithms. *Journal of Banking & Finance* 34, 11 (2010), 2767–2787.
[21] Diederik P Kingma and Jimmy Lei Ba. 2014. Adam: Amethod for stochastic optimization. In *Proc. 3rd Int. Conf. Learn. Representations.*
[22] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, Nov (2008), 2579–2605.
[23] Geoff N Masters. 1982. A Rasch model for partial credit scoring. *Psychometrika* 47, 2 (1982), 149–174.
[24] H Brendan McMahan. 2011. Follow-the-regularized-leader and mirror descent: Equivalence theorems and l1 regularization. (2011).
[25] Xiangfeng Meng, Yunhai Tong, Xinhai Liu, Yiren Chen, and Shaohua Tan. 2017. Netrating: Credit risk evaluation for loan guarantee chain in china. In *Pacific-Asia Workshop on Intelligence and Security Informatics.* Springer, 99–108.
[26] Xinhai Liu Xiangfeng Meng. 2015. Credit risk evaluation for loan guarantee chain in China. (2015).
[27] Yabo Ni, Dan Ou, Shichen Liu, Xiang Li, Wenwu Ou, Anxiang Zeng, and Luo Si. 2018. Perceive Your Users in Depth: Learning Universal User Representations from Multiple E-commerce Tasks. *arXiv preprint arXiv:1805.10727* (2018).
[28] Zhibin Niu, Dawei Cheng, Liqing Zhang, and Jiawan Zhang. 2018. Visual analytics for networked-guarantee loans risk management. In *2018 IEEE Pacific Visualization Symposium (PacificVis).* IEEE, 160–169.
[29] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 701–710.
[30] Pedro Ribeiro and Fernando Silva. 2012. Querying subgraph sets with g-tries. In *Proceedings of the 2nd ACM SIGMOD Workshop on Databases and Social Networks.* ACM, 25–30.
[31] Stan W Smith. 2010. An experiment in bibliographic mark-up: Parsing metadata for XML export. In *Proceedings of the 3rd. annual workshop on Librarians and Computers (LACâĂŽ10), Reginald N. Smythe and Alexander Noble (Eds.)*, Vol. 3. 422–431.
[32] Fei Tan, Xiurui Hou, Jie Zhang, Zhi Wei, and Zhenyu Yan. 2018. A deep learning approach to competing risks representation in peer-to-peer lending. *IEEE transactions on neural networks and learning systems* (2018).
[33] Veronique Van Vlasselaer, Jan Meskens, Dries Van Dromme, and Bart Baesens. 2013. Using social network knowledge for detecting spider constructions in social security fraud. In *Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on.* IEEE, 813–820.
[34] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2018. Graph attention networks. In *ICLR.*
[35] Shuai Xiao, Junchi Yan, Xiaokang Yang, Hongyuan Zha, and Stephen M Chu. 2017. Modeling the intensity function of point process via recurrent neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence.*
[36] Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining.* ACM, 425–434.
[37] Dingyuan Zhu, Peng Cui, Ziwei Zhang, Jian Pei, and Wenwu Zhu. 2018. High-order Proximity Preserved Embedding For Dynamic Networks. *IEEE Transactions on Knowledge and Data Engineering* (2018).