Transductive Semi-Supervised Metric Network for Reject Inference in Credit Scoring

Zhiyu Guo, Xiang Ao, *Member, IEEE*, and Qing He, *Member, IEEE*

***Abstract*— Credit scoring is an essential technique for credit risk management in the financial industry. However, most credit scoring models face the challenge of reject inference, which refers to the lack of post-loan performance data for rejected applicants, leading to sample selection bias and inaccurate credit assessment. Traditional credit scoring methods tackle this issue by assuming that the missing labels for rejected samples are missing at random (MAR) and by measuring sample similarity directly in the original feature space. Nevertheless, these strategies are not suitable for real-world business scenarios. Inspired by metric learning and transductive learning, we propose a novel credit scoring model called transductive semi-supervised metric network (TSSMN), which formalizes reject inference as a semi-supervised binary classification problem with the prior assumption of missing not at random (MNAR). TSSMN consists of two interconnected modules: the embedding metric network (EMN) that maps samples from the original feature space to the metric space for similarity measurement, and the transductive propagation network (TPN) that performs label propagation based on sample similarity. We evaluate TSSMN on a real-world credit dataset and compare it with traditional credit scoring methods. The results indicate that TSSMN can overcome sample selection bias and more accurately classify credit applicants. Therefore, TSSMN has the potential to enhance credit risk assessment in real-world business scenarios.**

***Index Terms*— Credit scoring, metric learning, reject inference, transductive learning.**

I. Introduction

With the rapid development of the Internet economy, there has been an increasing demand for personal microloans, resulting in the emergence of diverse online credit patterns [1]. In contrast to traditional banking institutions, Internet financial platforms use big data and machine learning techniques to develop intelligent financial risk control systems for identity verification, credit assessment, and loan decision-making [2]. This simplifies the loan process, enhances lending efficiency, and meets the growing demand for personal loans. However, risk control is a critical aspect of the financial industry, given the various challenges and potential risks it faces [3]. As such, both traditional financial institutions and emerging Internet financial platforms are seeking advanced methods to rapidly and accurately assess the creditworthiness of applicants and enable automated credit decisions [4]. Our research focuses on credit risk control [5], aiming to develop an effective model that enhances credit assessment accuracy, promotes rational credit decisions by financial platforms, and fosters stable and standardized development of the financial industry in the digital economy.

Credit scoring [6], [7], [8] is a widely used framework for credit risk control [9], as illustrated in Fig. 1. To evaluate creditworthiness based on information collected from applicants, financial platforms use credit scoring models constructed from historical data and expert experience. Applicants with higher scores are accepted for loans, while those with lower scores are rejected. Once loans are granted, the post-loan performance of accepted applicants is monitored, and those who repay on time are classified as "non-default applicants," whereas those who do not are classified as "default applicants." The original credit scoring model can be corrected and updated using information from default applicants. However, rejected applicants have no post-loan performance, making it difficult to correct the model's credit evaluation for them. Currently, most credit platforms label all rejected applicants as default applicants [10], resulting in biased estimation and the reject inference problem [11], [12]. Although many financial institutions have successfully used machine learning to develop credit scoring models with good results, the reject inference problem still lacks a satisfactory solution [13].

The objective of reject inference is to improve the credit scoring model by inferring the potential post-loan performance of rejected applicants [14]. Traditional credit scoring models assume that the missing labels of rejected applicants belong to missing at random (MAR) [15], but our experiments reveal that these assumptions do not align with actual business scenarios. To address this, we propose a semi-supervised [16] binary classification approach with the missing not at random (MNAR) assumption [15]. Precisely, we classify accepted applicants as either defaulters or non-defaulters, while rejected

Manuscript received 30 October 2022; revised 30 January 2023 and 16 March 2023; accepted 28 April 2023. Date of publication 25 May 2023; date of current version 3 April 2024. This work was supported in part by the National Key Research and Development Plan under Grant 2022YFC3303302, in part by the National Natural Science Foundation of China under Grant 61976204, and in part by the Alibaba Group through Alibaba Innovative Research Program. The work of Xiang Ao was supported by the Project of Youth Innovation Promotion Association Chinese Academy of Science (CAS), Beijing Nova Program under Grant Z201100006820062. *(Corresponding author: Xiang Ao.)*

Zhiyu Guo and Qing He are with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China, and also with the School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: guozhiyu22s@ict.ac.cn; heqing@ict.ac.cn).

Xiang Ao is with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China, also with the University of Chinese Academy of Sciences, Beijing 100049, China, and also with the Institute of Intelligent Computing Technology, Suzhou 215124, China (e-mail: aoxiang@ict.ac.cn).

Digital Object Identifier 10.1109/TCSS.2023.3276274

2329-924X © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

applicants are unlabeled and unevenly classified, with more potential defaulters than non-defaulters. Furthermore, unlike conventional credit scoring models directly using raw features and distance metrics to identify rejected applicants who resemble non-default ones, we embed the original features into the metric space before the similarity measurement. Our experiments show that this technique is more effective.

To address the unique challenges of reject inference and the limitations of existing methods, we propose a credit scoring model named transductive semi-supervised metric network (TSSMN). The model consists of two interconnected components: the embedding metric network (EMN) that maps samples from the original feature space to the metric space, and the transductive propagation network (TPN) that performs label propagation on an undirected graph. Under the MNAR assumption, TSSMN applies different propagation rules for accepted and rejected samples, respectively. We evaluated TSSMN on a real-world credit dataset and observed a significant improvement in accuracy compared to conventional methods, which demonstrates the efficacy of both modules for credit scoring tasks.

The main contributions of this work are concluded.

1) In contrast to traditional methods, our model incorporates the MNAR prior assumption. Experimental results demonstrate that this assumption improves classification accuracy, making our model more applicable to real-world business scenarios.
2) We provide mathematical proof that our proposed TSSMN is equivalent to the unbalanced label propagation algorithm (LPA). TSSMN surpasses LPA in terms of generalization capability and computational efficiency by utilizing an end-to-end regularization framework.
3) By jointly training the EMN and the TPN, TSSMN can better assess the similarity between applicants in the metric space, which can improve credit evaluation and enable reject inference. Our experiments on real-world data demonstrate its effectiveness.

The rest of this article is organized as follows. Section II reviews related work on credit scoring and reject inference. Section III details the credit scoring model based on a TSSMN. In Section IV, we experimentally demonstrate the effectiveness of TSSMN. Section V summarizes and discusses our work.

II. Related Work

Traditional credit scoring methods only consider accepted applicants, leading to biased parameter estimation and inaccurate credit assessment [14]. The problem of sample selection bias, which arises from the lack of post-loan performance for rejected applicants, can be considered a data-missing problem [15]. Feelders [17] identified three types of data missing in credit scoring scenarios: missing completely at random (MCAR), MAR, and MNAR. MCAR assumes that credit platforms randomly accept or reject loan applications, but this approach is not sustainable as it does not consider applicants' backgrounds and leads to severe losses [18]. Therefore, the MAR assumption is more commonly used, whereby credit platforms screen applications based on certain requirements

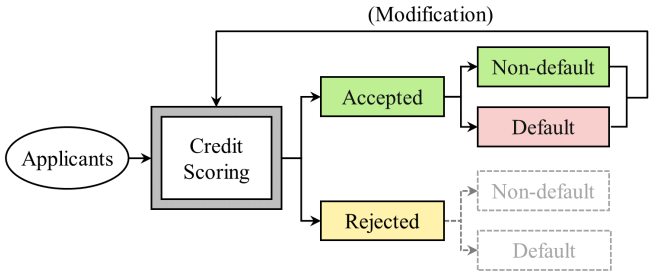

Fig. 1. Illustration of credit scoring and reject inference.

and accept or reject applicants accordingly. Under the MNAR assumption, credit decisions are influenced by both observed and unobserved variables [19]. To avoid bias in estimation results, it is necessary to introduce reject inference, i.e., adding rejected samples to the training data, which can significantly change model parameters and improve predicted results [14]. Therefore, it is inappropriate to ignore rejected samples in the construction of credit-scoring models.

In the early stages of research, various statistical methods were proposed for credit scoring and reject inference. Based on the MAR assumption, the re-weighting methods [20], [21] involved re-weighting accepted samples to represent the entire distribution, with weights typically determined by the probability of rejection/acceptance. Additionally, the fuzzy extension and parceling methods [14], [21], [22] partitioned applicants based on their credit scores, calculated the proportion of samples in different segments, and re-weighted them accordingly. Moreover, Feelders [17] used an expectation-maximization algorithm for reject the inference that pseudo-labels the rejected samples. In contrast, reject inference under the MNAR assumption has employed sophisticated statistical techniques such as augmentation and dilation. Banasik and Crook [23] conducted an in-depth study of these methods, incorporating both ideas into reject inference models. Despite this, the models' performance did not improve significantly. Besides, Bucker et al. [24] improved the reassignment-based reject inference approach, resulting in better classification results than the original credit scoring model.

In addition to statistical methods, machine learning methods have been developed and shown significant improvement in model performance. For example, in [25], the authors employ a self-training algorithm to enhance the performance of support vector machine (SVM) in credit scoring. Although the self-training algorithm is applied solely to improve SVM in [25], it can also enhance the performance of other classifiers like logistic regression (LR), multilayer perceptron (MLP), and extreme gradient boosting (XGB). Additionally, Li et al. [26] and Tian et al. [27] applied a semi-supervised support vector machine (S3VM) for reject inference. S3VM is trained using labeled and unlabeled samples to determine the supporting hyperplane, but it has a problem in fitting large-scale data. Recently, deep learning methods have developed rapidly with excellent performance in various application scenarios, and some researchers have proposed deep generative models for

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

reject inference in credit scoring. Mancisidor et al. [28] combined Bayesian and Gaussian mixture methods (GMMs) to construct generative models within a semi-supervised framework. Moreover, based on hidden Markov models and inspired by semi-supervised models, [29] is proposed for modeling biased credit scoring data with good results. These deep learning models perform better compared to traditional credit scoring models.

Generally, reject inference methods based on statistical analysis rely on the assumption of MAR, which assumes that both accepted and rejected samples have a similar pattern of post-loan performance [19]. However, it is not possible to theoretically verify this assumption due to the unavailability of actual post-loan performance data for rejected samples [14]. Although some semi-supervised learning methods [30], [31], [32] argue that the information from accepted samples can be used to infer the potential post-loan performance of rejected applicants, these methods are not widely accepted by the industry because inappropriate pseudo-labels may mislead the decision boundaries and perturb the stability of models [17]. Furthermore, deep generative models can effectively handle the missing data problem in rejecting inference. Nevertheless, some researchers [24] argue that generative models that use both labeled and unlabeled samples may cause classification performance to deteriorate when the underlying assumptions are inconsistent with reality.

III. Methodology

A. Notations

The set of all samples is denoted by $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$, where $x_i \in \mathbb{R}^m$ represents a feature vector, n is the number of samples, and m is the feature dimension. The sets of accepted and rejected samples are denoted by $\mathcal{X}_l = \{x_1, x_2, \ldots, x_{n_l}\}$, $\mathcal{X}_u = \{x_{n_l+1}, x_{n_l+2}, \ldots, x_{n_l+n_u}\}$, respectively, satisfying $\mathcal{X}_l \cup \mathcal{X}_u = \mathcal{X}$ and $\mathcal{X}_l \cap \mathcal{X}_u = \varnothing$. Here, n_l and n_u represent the number of accepted and rejected samples, respectively, and satisfy $n_l + n_u = n$. Accepted samples are labeled samples with post-loan performance, and the set of labels is denoted as $\mathcal{Y} = \{y_1, y_2, \ldots, y_{n_l}\} \in \{1, 2\}$, where $y_i = 1$ indicates a non-default (positive) sample and $y_i = 2$ indicates a default (negative) sample. Rejected samples do not have post-loan performance and are unlabeled samples. The objective of credit scoring is to construct and train a model f using the features of the full sample set $\mathcal{X}$ and the labels of accepted samples $\mathcal{Y}_l$, i.e., $(\mathcal{X}, \mathcal{Y}_l) \to f$, for evaluating the credit score (or the compliance probability) of new samples.

Rejected samples do not have post-loan performance data, and thus need reject inference to estimate their potential post-loan performance. The absence of labels for rejected samples leads to the issue of missing data, which is typically categorized into three types: MCAR, MAR, and MNAR. Let $s_i \in \{1, 2\}$ denote the credit decision for sample i, where $s_i = 1$ indicates acceptance and $s_i = 2$ indicates rejection. Therefore, accepted samples $x_i \in \mathcal{X}_l$ satisfy $s_i = 1$ and have label information, while rejected samples $x_i \in \mathcal{X}_u$ satisfy $s_i = 2$ and lack label information. Traditional credit scoring methods assume that the label missing of rejected samples belongs to either MCAR or MAR, which can be expressed mathematically as

$$\text{MCAR: } p(s \mid x, y) = p(s \mid x) = p(s) \tag{1}$$

$$\text{MAR: } p(s \mid x, y) = p(s \mid x) \neq p(s). \tag{2}$$

Under these assumptions, accepted samples can represent the overall distribution of samples, and reject inference is unnecessary. Therefore, only labeled accepted samples are utilized for supervised learning to construct the credit scoring model, i.e., $(\mathcal{X}_l, \mathcal{Y}_l) \to f$. However, in the real business scenario of credit scoring, the original model based on historical experience is accurate for most credit decisions. That is, among rejected samples, there are significantly more potential default samples than potential non-default samples, and the label missing of rejected samples should belong to MNAR, mathematically expressed as

$$\text{MNAR: } p(s \mid x, y) \neq p(s \mid x) \neq p(s). \tag{3}$$

The set of potential labels for rejected samples is denoted by $\mathcal{Y}_u = \{y_{n_l+1}, \ldots, y_{n_l+n_u}\} \in \{1, 2\}$, where $y_i = 1$ indicates a potential non-default sample and $y_i = 2$ indicates a potential default sample. Given the MNAR assumption, we have

$$\sum_{i=n_l+1}^{n_l+n_u} \mathbb{I}(y_i = 2) > \sum_{i=n_l+1}^{n_l+n_u} \mathbb{I}(y_i = 1) \tag{4}$$

where $\mathbb{I}(a)$ is the indicator function, with $\mathbb{I}(a) = 1$ if a is true and $\mathbb{I}(a) = 0$ otherwise. In real-world credit scoring scenarios, accepted samples exhibit bias compared to the overall sample distribution. Therefore, a model based solely on biased samples is inadequate to make accurate predictions for samples from the general population. Consequently, rejecting inference is necessary. A reasonable credit scoring model f should employ both labeled accepted samples and unlabeled rejected samples to learn while considering the MNAR assumption, i.e., $(\mathcal{X}, \mathcal{Y}_l, \mathcal{Y}_u) \to f$.

B. Transductive Propagation Network

Semi-supervised methods improve the generalization performance of models by using both labeled and unlabeled samples, which are suitable for credit scoring. The LPA [32] is a classical semi-supervised transductive learning technique that assumes samples with similar characteristics have the same labels. LPA constructs an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ using both labeled and unlabeled samples, where nodes represent samples and edges represent the similarities between them. The algorithm then iteratively propagates labels along the edges of the graph until convergence is reached, resulting in the predicted labels of samples. LPA typically employs the Gaussian similarity function to measure the similarities between samples

$$W_{ij} = \begin{cases} \exp\left(-\|x_i - x_j\|/2\sigma^2\right), & i \neq j \\ 0, & i = j \end{cases} \tag{5}$$

where $\sigma > 0$ is the scale parameter. The similarity matrix W is constructed using the Gaussian similarity function and symmetrically normalized to produce the normalized similarity

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

matrix $S = D^{-1/2}WD^{-1/2}$, where D is a diagonal matrix with $D_{ii} = \sum_{j=1}^{n} W_{ij}$. Let $\mathcal{F}$ be the set of all $n \times 2$ non-negative real matrices, and let $Y \in \mathcal{F}$ be the one-hot encoding matrix of labels, where labeled samples are encoded and unlabeled samples are set to zero

$$Y_{ij} = \begin{cases} 1, & x_i \in \mathcal{X}_l \text{ and } y_i = j \\ 0, & \text{otherwise}, \quad i = 1, 2, \ldots, n. \end{cases} \tag{6}$$

Let $F^{(t)} \in \mathcal{F}$ denote the predicted label matrix from LPA after t rounds of iteration. The label propagation process is an iterative computation based on the following recursive equation:

$$F^{(0)} = Y \tag{7}$$

$$F^{(t+1)} = \alpha S F^{(t)} + (1-\alpha)Y \tag{8}$$

where $SF^{(t)}$ denotes the aggregated label information from similar samples, Y denotes the initial label information, and $\alpha \in (0, 1)$ is the aggregation weight that represents the relative amount of aggregated label information from similar samples and samples themselves during iterations. The iterative calculation of LPA has been proven [32] to be convergent

$$F^* = \lim_{t\to\infty} F^{(t)} = (1-\alpha)(I - \alpha S)^{-1} Y. \tag{9}$$

This convergence directly leads to the predicted labels for the samples

$$\mathcal{Y}^* = \left\{ \hat{y}_i \mid \hat{y}_i = \arg\max_j F^*_{ij}; i = 1, 2, \ldots, n \right\}. \tag{10}$$

As a transductive learning method, LPA can only be trained and used for prediction within a specific sample set $\mathcal{X}$, i.e., $(\mathcal{X}, \mathcal{Y}_l) \to \mathcal{Y}^*$, and cannot be extended to the entire sample space like inductive learning methods. However, the ultimate goal of credit scoring is to obtain an inductive model f that applies to new samples. One simple idea is to first obtain the predicted labels $\mathcal{Y}^*$ for the full set of samples using LPA and then use supervised learning with the total set of labeled samples to obtain the final credit scoring model f, i.e., $(\mathcal{X}, \mathcal{Y}_l) \to (\mathcal{X}, \mathcal{Y}^*) \to f$. However, LPA is poorly integrated with the supervised model, and the supervised information is not sufficient to guide the entire learning process for effective end-to-end learning [33]. In comparison with modular models, the end-to-end learning paradigm can fully utilize supervised information and has low engineering complexity, making it easier for direct deployment and application in real business scenarios for credit scoring.

LPA has the advantageous property [34] that the iterative solution of the label matrix F^* via label propagation is equivalent to solving the following optimization problem:

$$\mathcal{L}_F = \mathcal{L}_{\text{SMO}} + \lambda \mathcal{L}_{\text{FIT}} \tag{11}$$

$$\mathcal{L}_{\text{SMO}} = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} W_{ij} \left\| \frac{F_i}{\sqrt{D_{ii}}} - \frac{F_j}{\sqrt{D_{jj}}} \right\|^2 \tag{12}$$

$$\mathcal{L}_{\text{FIT}} = \sum_{i=1}^{n} \| F_i - Y_i \|^2 \tag{13}$$

where λ is the weight coefficient satisfying $\lambda = 1/\alpha - 1$, and $\alpha \in (0, 1)$ is the aggregation weight in LPA. The loss function $\mathcal{L}_F$ is defined as the sum of two terms: $\mathcal{L}_{\text{SMO}}$, which represents the smoothing constraint loss and ensures that predicted labels are consistent with those of similar samples, and $\mathcal{L}_{\text{FIT}}$, which represents the fitting constraint loss and ensures that predicted labels are consistent with the initial labels. F_i can be considered as the model f_φ defined on the sample space, where φ denotes the model's parameters. By rewriting (11)–(13) as matrix operations, the model f_φ can be trained using the following regularization framework:

$$\begin{aligned} \mathcal{L}_f &= \mathcal{L}_{\text{SMO}} + \lambda \mathcal{L}_{\text{FIT}} \\ &= f_\varphi^T (L + \lambda I) f_\varphi - 2\lambda Y^T f_\varphi + \lambda Y^T Y \end{aligned} \tag{14}$$

where $\lambda > 0$ is the weight coefficient, $L = I - D^{-1/2}WD^{-1/2}$ is the symmetric normalized Laplacian matrix, and f_φ is the credit scoring model using an MLP with a 2-D softmax function in the output layer. By using gradient approaches, the loss function $\mathcal{L}_f$ can be iteratively optimized with sample data, enabling end-to-end learning to obtain the credit scoring model f, i.e. $(\mathcal{X}, \mathcal{Y}_l, f_\varphi) \to f$. The output of model f is the prediction of probabilities, given by $f(x_i) = [f_1(x_i), f_2(x_i)]$, where $f_1(x_i)$ represents the non-default probability and $f_2(x_i)$ represents the default probability, satisfying $f_1(x_i) + f_2(x_i) = 1$. Finally, the model's predicted sample labels are obtained from the following equation:

$$\mathcal{Y}^* = \left\{ \hat{y}_i \mid \hat{y}_i = \arg\max_j f_j(x_i); i = 1, 2, \ldots, n \right\}. \tag{15}$$

The above regularization framework based on LPA is called TPN, or TPN for short.

C. TPN Under the MNAR Assumption

Some existing methods [30], [35] make use of LPA or TPN directly for credit scoring. However, these methods only consider the label information of accepted samples and the relationships between samples when predicting the labels of rejected samples. They do not consider the MNAR assumption in credit scoring scenarios, which posits that there are typically more potential default samples than potential non-default samples among rejected samples. Therefore, we propose the following improvements to LPA to address the particularities of credit scoring problems.

1) In the initial state, rejected samples are considered as potential default samples, i.e., $y_{n_l+1} = y_{n_l+2} = \cdots = y_{n_l+n_u} = 2$. The one-hot encoding matrix Y is modified as follows:

$$Y_{ij} = \begin{cases} 1, & y_i = j \\ 0, & \text{otherwise} \end{cases} \qquad i = 1, 2, \ldots, n. \tag{16}$$

2) The relative amounts of label information aggregated by accepted and rejected samples from similar samples and samples themselves should be different during label propagation

$$F^{(0)} = Y \tag{17}$$

$$F^{(t+1)} = PSF^{(t)} + (I - P)Y \tag{18}$$

$$P = \begin{bmatrix} \alpha I_l & 0 \\ 0 & \beta I_u \end{bmatrix} \tag{19}$$

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

where $\alpha, \beta \in (0, 1)$ are aggregation weights for accepted and rejected samples, respectively, and satisfy $\alpha < \beta$. During label propagation, accepted samples are more influenced by their initial labels, whereas rejected samples are more influenced by labels of similar samples. I_l and I_u are the identity matrices of size $n_l \times n_l$ and $n_u \times n_u$, respectively, which form the diagonal matrix P. The proposed method, which is an improved LPA based on the MNAR assumption, is referred to as MNAR-LPA.

The iterative computation of MNAR-LPA remains convergent, as demonstrated by the following proof:

$$\begin{aligned} F^* &= \lim_{t\to\infty} F^{(t)} \\ &= \lim_{t\to\infty} (PS)^{t-1}Y + (I-P)\lim_{t\to\infty}\sum_{i=0}^{t-1}(PS)^i Y \\ &= (I-P)(I-PS)^{-1}Y. \end{aligned} \tag{20}$$

Likewise, the iterative solution of the label matrix F^* using MNAR-LPA is equivalent to solving the following optimization problem:

$$\mathcal{L}_F = \mathcal{L}_{\text{SMO}} + \lambda\mathcal{L}_{\text{FIT}} + \mu\mathcal{L}_{\text{MNAR}} \tag{21}$$

$$\mathcal{L}_{\text{SMO}} = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} W_{ij}\left\|\frac{F_i}{\sqrt{D_{ii}}} - \frac{F_j}{\sqrt{D_{jj}}}\right\|^2 \tag{22}$$

$$\mathcal{L}_{\text{FIT}} = \sum_{i=1}^{n_l}\|F_i - Y_i\|^2 \tag{23}$$

$$\mathcal{L}_{\text{MNAR}} = \sum_{j=n_l+1}^{n_l+n_u}\|F_j - Y_j\|^2. \tag{24}$$

A brief proof is given below. Rewriting (21)–(24) into the form of matrix operations, we have

$$\mathcal{L}_F = F^T(I + C - S)F - 2Y^TCF + Y^TCY \tag{25}$$

$$C = \begin{bmatrix} \lambda I_l & 0 \\ 0 & \mu I_u \end{bmatrix} \tag{26}$$

where S is the symmetric normalization matrix of W, i.e., $S = D^{-1/2}WD^{-1/2}$. The optimal value point F^* that minimizes the loss function $\mathcal{L}_F$ satisfies

$$\begin{aligned} \left.\frac{\partial\mathcal{L}_F}{\partial F}\right|_{F^*} &= 2(I + C - S)F^* - 2CY = 0 \\ F^* &= C(I + C - S)^{-1}Y \\ &= \left(I - (I+C)^{-1}\right)\left(I - (I+C)^{-1}S\right)^{-1}Y. \end{aligned} \tag{27}$$

Let $P = (I + C)^{-1}$, i.e., $\lambda = 1/\alpha - 1$ and $\mu = 1/\beta - 1$, where $\alpha, \beta \in (0, 1)$ are the aggregation weights of accepted and rejected samples in MNAR-LPA, respectively, then F^* can be rewritten as

$$F^* = (I - P)(I - PS)^{-1}Y. \tag{28}$$

We observe that F^* is equivalent to the closed solution of MNAR-LPA calculated by (20), confirming that MNAR-LPA has a regularization framework similar to that of TPN. Consequently, we can train the credit scoring model f_φ using the following regularization framework:

$$\begin{aligned} \mathcal{L}_f &= \mathcal{L}_{\text{SMO}} + \lambda\mathcal{L}_{\text{FIT}} + \mu\mathcal{L}_{\text{MNAR}} \\ &= f_\varphi^T(L + C)f_\varphi - 2Y^TCf_\varphi + Y^TCY \end{aligned} \tag{29}$$

where $\lambda > \mu > 0$ are weight coefficients, C is the coefficient matrix, and L is the Laplacian matrix. The loss function $\mathcal{L}_f$ comprises three terms: the first term $\mathcal{L}_{\text{SMO}}$ is the smoothing constraint loss; the second term $\mathcal{L}_{\text{FIT}}$ is the fitting constraint loss on accepted samples; and the third term $\mathcal{L}_{\text{MNAR}}$ is the fitting constraint loss on rejected samples with the MNAR assumption. We refer to this regularization framework as the TPN under the MNAR assumption, or MNAR-TPN for brevity.

D. MNAR-TPN With Metric Learning

LPA assumes that the original feature space of samples is the metric space, which enables the similarity metric to be performed directly on the original feature space, typically using the Gaussian similarity function. However, this metric can only measure the linear similarity between original features. In practical business scenarios, the original features usually have varying magnitudes and nonlinear correlations. Therefore, using the original features of samples for similarity metrics is unsuitable. According to the related study of manifold learning [36], we can set up a mapping function, denoted as g, which maps samples from the original feature space $\mathcal{X}$ to the metric space $\mathcal{M}$, i.e., $\mathcal{X} \xrightarrow{g} \mathcal{M}$. After mapping, the similarity between samples can be measured using the Gaussian similarity function

$$W_{ij} = \begin{cases} \exp\left(-\left\|g(x_i) - g(x_j)\right\|/2\sigma^2\right), & i \neq j \\ 0, & i = j. \end{cases} \tag{30}$$

Since the mapping function g is typically not directly available, it is necessary to learn a mapping function g_ϕ that contains parameters ϕ. This process is commonly known as metric learning [35]. In the metric learning framework, the Laplacian matrix L in the loss function of MNAR-TPN is rewritten as L_ϕ with respect to the parameters ϕ

$$\mathcal{L}_{f,g} = f_\varphi^T\left(L_\phi + C\right)f_\varphi - 2Y^TCf_\varphi + Y^TCY \tag{31}$$

where $L_\phi = I - D_\phi^{-1/2}W_\phi D_\phi^{-1/2}$ is the symmetric normalized Laplacian matrix generated by the mapping g_ϕ, and D_ϕ is the diagonal matrix satisfying $(D_\phi)_{ii} = \sum_{j=1}^{n}(W_\phi)_{ij}$. Based on the proof of (27), for any symmetric normalized Laplacian matrix L_ϕ, there exists $f_\varphi = C(L_\phi + C)^{-1}Y$ such that the loss function $\mathcal{L}_{f,g}$ takes the minimum value. Therefore, f_φ can be regarded as a function on L_ϕ and substituted into the loss function $\mathcal{L}_{f,g}$ to derive the loss function $\mathcal{L}_g$

$$\mathcal{L}_g = \mathcal{L}_{f,g}\big|_{f_\varphi = C(L_\phi + C)^{-1}Y}. \tag{32}$$

The process of metric learning is to minimize the following loss function $\mathcal{L}_g$ using sample data:

$$\mathcal{L}_g = -Y^TC^2\left(L_\phi + C\right)^{-1}Y + Y^TCY. \tag{33}$$

According to the metric learning research [37], the mapping g_ϕ can be implemented using an MLP with a lower output dimension than the input dimension. Using the metric learning

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

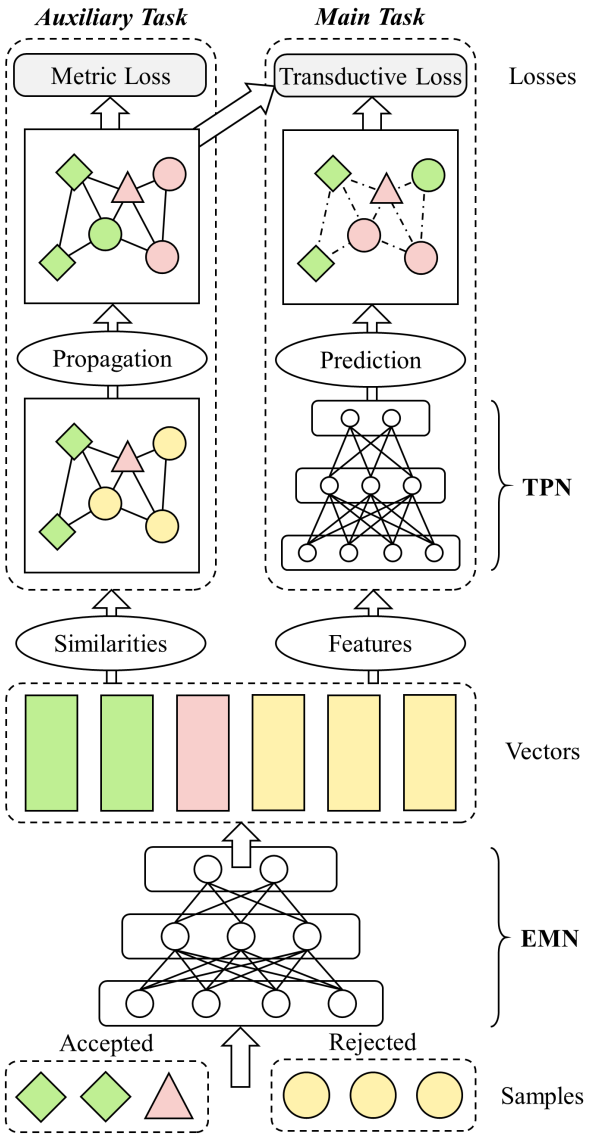

Fig. 2. Architecture overview of TSSMN for reject inference in credit scoring. TSSMN mainly consists of two sub-networks: EMN maps samples from the original feature space to the metric space for similarity measurement; TPN performs label propagation based on the sample similarity.

framework, the mapping g is obtained to map the samples $\mathcal{X}$ to the metric space $\mathcal{M}$, and the similarity matrix $W_{\mathcal{M}}$ is constructed. The symmetric normalization is then applied to obtain $S_{\mathcal{M}}$, which is computed by $S_{\mathcal{M}} = D_{\mathcal{M}}^{-1/2} W_{\mathcal{M}} D_{\mathcal{M}}^{-1/2}$. Subsequently, the Laplacian matrix $L_{\mathcal{M}}$ is calculated by $L_{\mathcal{M}} = I - S_{\mathcal{M}}$. Finally, the credit scoring model f_φ is trained using the MNAR-TPN regularization framework

$$\begin{aligned}\mathcal{L}_f &= \mathcal{L}_{\text{SMO}} + \lambda \mathcal{L}_{\text{FIT}} + \mu \mathcal{L}_{\text{MNAR}} \\ &= f_\varphi^T (L_{\mathcal{M}} + C) f_\varphi - 2Y^T C f_\varphi + Y^T C Y. \end{aligned} \tag{34}$$

The above regularization framework, which improves MNAR-TPN with metric learning, is called MNAR-TMN.

E. Transductive Semi-Supervised Metric Network

Studies on metric learning [38] indicate that classification becomes easier after mapping samples from the original feature space to the metric space, as similar samples are brought closer while dissimilar samples are pushed apart. Thus, after obtaining the mapping g through metric learning, the classification model f_φ learns in the metric space $\mathcal{M}$ instead of the original feature space $\mathcal{X}$

$$f_\varphi(\mathcal{X}) \to f_\varphi(g(\mathcal{X})) \to f_\varphi(\mathcal{M}). \tag{35}$$

Furthermore, the loss functions of transductive learning $\mathcal{L}_f$ and metric learning $\mathcal{L}_g$ are combined into a unified loss function $\mathcal{L}$. This approach enables the simultaneous execution of transductive learning and metric learning

$$\mathcal{L} = \mathcal{L}_f + \eta \mathcal{L}_g \tag{36}$$

$$\mathcal{L}_f = f_\varphi^T \left(L_\phi + C\right) f_\varphi - 2Y^T C f_\varphi \tag{37}$$

$$\mathcal{L}_g = -Y^T C^2 \left(L_\phi + C\right)^{-1} Y \tag{38}$$

where η is a hyperparameter. In addition, the constant terms in both $\mathcal{L}_f$ and $\mathcal{L}_g$ are eliminated. To enhance the performance of MNAR-TPN, we introduce an end-to-end regularization learning framework called TSSMN. Within this framework, the mapping function g_ϕ not only provides objective similarity measures but also generates embedded features that are easily classifiable for the model f_φ. Thus, we refer to the mapping function g_ϕ as the EMN and the model f_φ as the TPN. $\mathcal{L}_f$ is called transductive loss and $\mathcal{L}_g$ is called metric loss. Fig. 2 provides an overview of the TSSMN architecture.

As the construction of the similarity matrix and the calculation of the inverse matrix have a time complexity of $O(n^3)$ during training, we limit the number of samples used for building subgraphs by selecting a subset of samples $\mathcal{X}_s \subset \mathcal{X}$ in each round. Additionally, to prevent learning bias caused by category imbalance, we ensure that the numbers of non-default, default, and rejected samples are balanced when sampling. In the credit scenario, default samples are much less common than non-default and rejected samples. Therefore, we repeatedly sample from the default samples in each round to ensure an adequate representation of default samples.

IV. Experiments

A. Dataset

The dataset used in the experiment is obtained from the Rong360 financial platform, including 100 000 samples, with 30 465 accepted and 69 535 rejected samples. Among the accepted samples, 28 628 are non-default (positive) and 1837 are default (negative). Furthermore, 3000 rejected samples are accepted by the Rong360 financial platform, and their post-loan performance is collected. Among these, 2639 are non-default and 361 are default. These rejected samples are utilized to estimate the effectiveness of reject inference and are not included as labeled samples in the model's training process.

B. Metrics

Accuracy is the most commonly used model evaluation index, which measures the proportion of correctly classified samples out of all samples. However, it is not well-suited

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

TABLE I
COMPARISON OF PERFORMANCE BETWEEN TRADITIONAL CREDIT SCORING METHODS AND OUR PROPOSED METHOD ON THE RONG360 CREDIT DATASET, EVALUATED BY AUC(%), KS(%), AUC+(%), KS+(%), AND AR(%). AVERAGE VALUES ARE LISTED

Type	Model	Metric				
		AUC↑	KS↑	AUC+↑	KS+↑	AR↓
Baseline	$LR^{(A)}$	77.65	43.12	75.18	41.35	65.79
	$MLP^{(A)}$	79.30	47.13	78.52	45.49	86.48
	$SVM^{(A)}$	78.92	46.09	77.18	42.91	63.47
	$LR^{(AR)}$	69.83	33.68	69.36	33.60	19.02
	$MLP^{(AR)}$	77.52	43.96	75.21	41.91	15.75
	$SVM^{(AR)}$	74.89	40.30	71.37	38.09	16.84
Semi-Supervised Learning	LPA+LR	78.13	44.37	76.98	43.01	58.12
	LPA+MLP	80.71	48.76	79.82	46.52	69.78
	LPA+SVM	80.82	48.77	80.04	46.70	53.13
	S3VM	81.89	50.82	80.71	48.86	52.67
	TSVM	81.56	50.34	80.20	48.28	43.12
Ablation	TPN	80.93	48.81	78.95	46.83	67.49
	MNAR-TPN	81.04	48.94	79.34	47.73	16.87
	MNAR-TMN	82.17	51.24	81.02	49.81	15.60
	TSSMN	**82.43**	**51.70**	**81.57**	**50.29**	**15.34**

for credit scoring problems in which default samples heavily outnumber non-default samples. A model that accurately identifies only non-default samples may have a high accuracy score but fails to meet risk control requirements. Thus, we use Kolmogorov–Smirnov statistic (KS) and area under curve (AUC), widely used model evaluation metrics in credit scoring, to measure model performance. KS is a metric that relates to the true positive rate (TPR) and the false positive rate (FPR). The TPR, also known as recall, indicates the model's ability to identify positive samples

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{39}$$

where TP and FN denote the number of true positive and false negative samples, respectively. FPR indicates the fraction of negative samples that the model wrongly predicts as positive

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{40}$$

where FP and TN denote the number of false positive and true negative samples, respectively. KS measures the largest gap between the TPR and FPR across various classification thresholds

$$\mathrm{KS} = \max(\mathrm{TPR} - \mathrm{FPR}). \tag{41}$$

The KS metric compares the cumulative distributions of positive and negative samples, with a larger KS value indicating a better ability of the model to classify samples. AUC is the area under the receiver operating characteristic (ROC) curve, which plots FPR on the horizontal axis and TPR on the vertical axis, and each point on the curve represents a classification threshold. The higher the AUC value, the better the model's performance.

The AUC and KS metrics are limited in that they only evaluate the model's performance on accepted samples, without accounting for rejected samples and the model's reject inference ability. Since the Rong360 dataset includes 3000 labeled rejected samples, we add them to the test set to construct the AUC+ and KS+ metrics, which provide a more comprehensive evaluation of the model's performance on both accepted and rejected samples. Additionally, we propose the accept-rejection rate (AR) as another crucial metric. AR measures the proportion of predicted non-default samples among the rejected samples relative to the total number of rejected samples. Lower values of AR indicate better model performance in identifying potential non-defaulters among rejected samples.

C. Compared Methods

We compare 15 methods in our experiments. $LR^{(A)}$, $MLP^{(A)}$, $SVM^{(A)}$, $LR^{(AR)}$, $MLP^{(AR)}$, and $SVM^{(AR)}$ are baseline methods for credit scoring. LPA + LR, LPA + MLP, LPA + SVM, S3VM, and TSVM are classic semi-supervised methods. TPN, MNAR-TPN, MNAR-TMN, and TSSMN are our proposed methods. In credit scoring, there are two common model training strategies: one is to use only accepted samples for supervised learning; the other is to label all rejected samples as default samples and use all samples for supervised learning. We adopt both strategies in our experiments and denote them as (A) and (AR). Table I shows the experimental results.

D. Comparative Analysis of Multiple Methods

Firstly, we compare three widely used models: LR, MLP, and SVM. Since reject inference is not considered, $LR^{(A)}$ performs much worse on rejected samples than on accepted samples. $MLP^{(A)}$ and $SVM^{(A)}$ show performance improvement compared to $LR^{(A)}$, indicating that the credit scoring dataset is not simply linearly separable. By using rejected samples marked as default samples for training, the AR metrics of $LR^{(AR)}$, $MLP^{(AR)}$, and $SVM^{(AR)}$ decrease significantly and meet the risk control requirements. However, modeling with

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

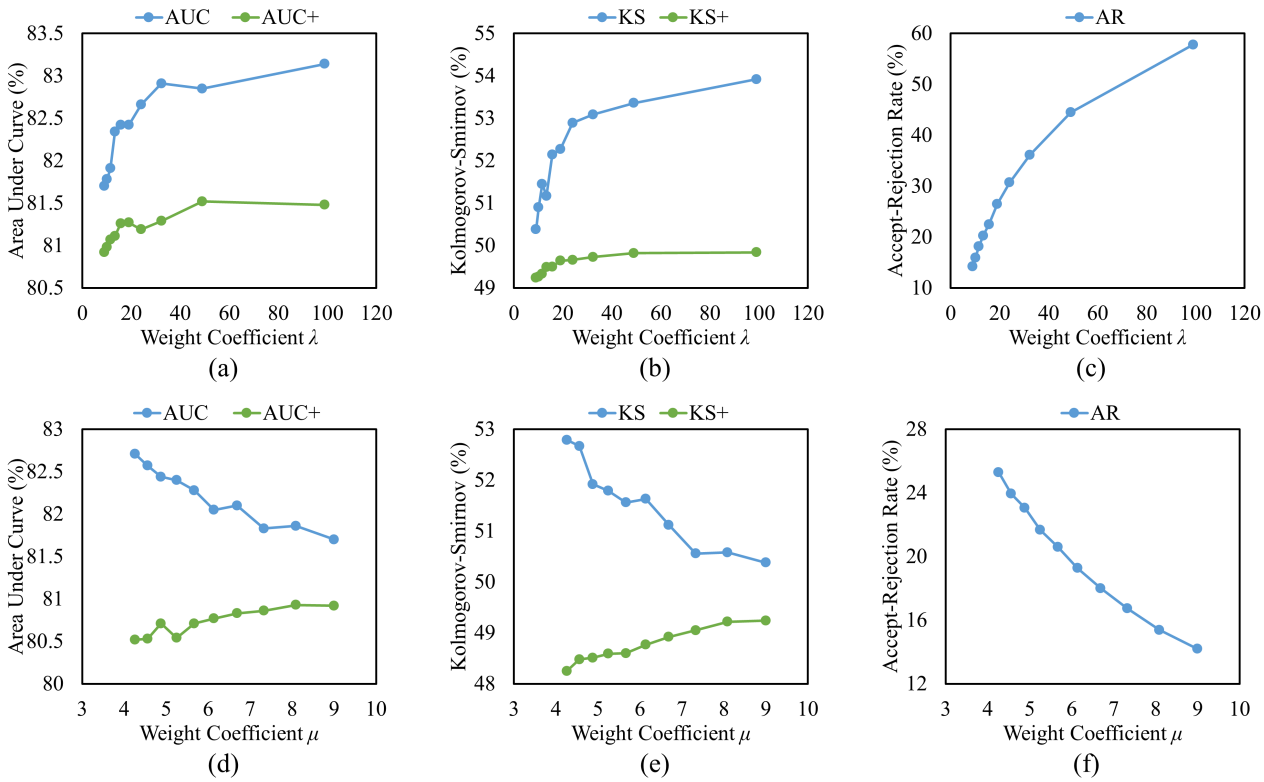

Fig. 3. Performance of TSSMN on the dataset with varying hyper-parameters. (a)–(c) Impact of the weight coefficient λ. (d)–(f) Impact of the weight coefficient μ on AUC, AUC+, KS, KS+, and AR.

both types of samples results in a more complex training set distribution and leads to varying degrees of performance decrease.

Some credit scoring methods employ the LPA to pseudo-label rejected samples and use both accepted and rejected samples for supervised learning. These methods, namely LPA + LR, LPA + MLP, and LPA + SVM, can enhance the model's performance compared to models that solely utilize accepted samples. Although more advanced semi-supervised learning methods, such as S3VM and transductive SVM (TSVM), outperform LR, MLP, and SVM, they do not address the classification imbalance problem of rejected samples. Consequently, these methods still have high AR metrics and do not meet the business requirements of credit risk control.

E. Ablation Test

By fusing LPA and MLP, we obtain an end-to-end framework called TPN. The performance of TPN is only slightly better than that of LPA + MLP, while the AR metric is still high because TPN does not consider the class imbalance problem of rejected samples either. We improve TPN by introducing the MNAR assumption, i.e., the proportion of potential non-default samples is lower than default samples among rejected samples. The improved model is referred to as MNAR-TPN. MNAR-TPN maintains the model's classification performance while significantly reducing the AR metric. The model predicts only 16.87% of rejected samples as potential non-default samples, which meets the credit scoring requirements.

We also introduce metric learning on top of MNAR-TPN, and the improved model is referred to as MNAR-TMN. With metric learning, the model's performance is significantly improved, and the AR metric remains low, demonstrating that metric learning can measure complex similarity relationships between samples and applies to different credit scoring tasks. We further upgrade MNAR-TPN to TSSMN, which has a further improved performance and a low AR metric, demonstrating the effectiveness of transductive learning and metric learning being performed simultaneously.

To summarize, traditional credit scoring methods struggle to balance the model's performance and the degree of mining rejected samples. The business scenario of credit scoring requires a high classification performance for accepted samples and a desire to reduce the proportion of rejected samples classified as potential non-default samples. Through experiments, we demonstrate that the proposed model outperforms traditional credit scoring methods and significantly reduces the proportion of rejected samples mined by the model. In addition, we evaluate each improved part of the model separately, which is equivalent to conducting ablation experiments. The results show that each part improves the model's performance to some degree, thus proving the effectiveness of the proposed method.

F. Parameter Sensitivity Analysis

The model's two main parameters, λ and μ, impact the prediction of accepted and rejected samples, respectively. A higher λ value ensures the predicted labels of accepted samples are closer to the real labels, while a larger μ value

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

requires the model to classify rejected samples as potential defaults, consistent with the MNAR assumption. We use more intuitive parameters α and β and their relationships with λ and μ ($\lambda = 1/\alpha - 1$ and $\mu = 1/\beta - 1$, where $\alpha < \beta$) to train the model and evaluate experimental results. The results are shown in Fig. 3.

In the first experiment, we fix $\beta = 0.1$ ($\mu = 9$) and vary α to obtain corresponding λ values. The results indicate that as λ increases, model performance improves for accepted samples but declines for rejected samples. A larger λ requires accepted samples' predicted labels to be closer to the real labels, and the MNAR assumption's impact on the model decreases. Therefore, λ should not be too large.

In the second experiment, we set $\alpha = 0.1$ ($\lambda = 9$) and vary β to obtain corresponding μ values. As μ increases, model performance deteriorates for accepted samples but improves for rejected samples. A larger μ requires the model to classify rejected samples as potential default samples, in line with the MNAR assumption. Thus, μ should not be too low.

V. Conclusion and Discussion

Credit scoring is a crucial technique for credit risk management, yet it faces the challenge of the reject inference problem. In this article, we introduce a novel credit scoring model, named TSSMN, which formalizes reject inference as a semi-supervised binary classification problem under the MNAR assumption. TSSMN comprises an EMN and a TPN, integrating metric learning and transductive learning. Our experimental results reveal that TSSMN outperforms traditional methods in reject inference and credit assessment, thus illustrating that the MNAR assumption, metric learning, and transductive learning are effective strategies for credit scoring. These strategies can inspire the development of more efficient credit scoring models. In future work, we aim to extend this method to time series and graph transaction data, thereby enabling reject inference in more complex financial scenarios.

Acknowledgment

The authors would like to thank the Anonymous Reviewers and the Associate Editors for their review efforts.

References

[1] S. Han, K. Zhu, M. Zhou, and X. Cai, "Information-utilization-method-assisted multimodal multiobjective optimization and application to credit card fraud detection," *IEEE Trans. Computat. Social Syst.*, vol. 8, no. 4, pp. 856–869, Aug. 2021.

[2] R. Li, Z. Liu, Y. Ma, D. Yang, and S. Sun, "Internet financial fraud detection based on graph learning," *IEEE Trans. Computat. Social Syst.*, early access, Jul. 15, 2022, doi: 10.1109/TCSS.2022.3189368.

[3] M. R. Machado and S. Karray, "Assessing credit risk of commercial customers using hybrid machine learning algorithms," *Expert Syst. Appl.*, vol. 200, Aug. 2022, Art. no. 116889.

[4] N. Bussmann, P. Giudici, D. Marinelli, and J. Papenbrock, "Explainable machine learning in credit risk management," *Comput. Econ.*, vol. 57, no. 1, pp. 203–216, Jan. 2021.

[5] F. Shen, X. Zhao, G. Kou, and F. E. Alsaadi, "A new deep learning ensemble credit risk evaluation model with an improved synthetic minority oversampling technique," *Appl. Soft Comput.*, vol. 98, Jan. 2021, Art. no. 106852.

[6] L. Zheng, G. Liu, C. Yan, C. Jiang, M. Zhou, and M. Li, "Improved TrAdaBoost and its application to transaction fraud detection," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 5, pp. 1304–1316, Oct. 2020.

[7] Z. Li, G. Liu, and C. Jiang, "Deep representation learning with full center loss for credit card fraud detection," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 2, pp. 569–579, Apr. 2020.

[8] Y. Xie, G. Liu, C. Yan, C. Jiang, and M. Zhou, "Time-aware attention-based gated network for credit card fraud detection by extracting transactional behaviors," *IEEE Trans. Computat. Social Syst.*, early access, Mar. 30, 2022, doi: 10.1109/TCSS.2022.3158318.

[9] N. Kozodoi, J. Jacob, and S. Lessmann, "Fairness in credit scoring: Assessment, implementation and profit implications," *Eur. J. Oper. Res.*, vol. 297, no. 3, pp. 1083–1094, Mar. 2022.

[10] T. Li, G. Kou, Y. Peng, and P. S. Yu, "An integrated cluster detection, optimization, and interpretation approach for financial data," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13848–13861, Dec. 2022.

[11] F. Shen, X. Zhao, and G. Kou, "Three-stage reject inference learning framework for credit scoring using unsupervised transfer learning and three-way decision theory," *Decis. Support Syst.*, vol. 137, Oct. 2020, Art. no. 113366.

[12] Q. Liu et al., "RMT-Net: Reject-aware multi-task network for modeling missing-not-at-random data in financial credit scoring," *IEEE Trans. Knowl. Data Eng.*, early access, May 30, 2022, doi: 10.1109/TKDE.2022.3179025.

[13] N. Kozodoi, P. Katsas, S. Lessmann, L. Moreira-Matias, and K. Papakonstantinou, "Shallow self-learning for reject inference in credit scoring," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, 2019, pp. 516–532.

[14] A. Ehrhardt, C. Biernacki, V. Vandewalle, P. Heinrich, and S. Beben, "Reject inference methods in credit scoring," *J. Appl. Statist.*, vol. 48, no. 13, pp. 2734–2754, 2021.

[15] C. K. Enders, *Applied Missing Data Analysis*. New York, NY, USA: Guilford Press, 2010.

[16] X. Yang, Z. Song, I. King, and Z. Xu, "A survey on deep semi-supervised learning," *IEEE Trans. Knowl. Data Eng.*, early access, Nov. 8, 2022, doi: 10.1109/TKDE.2022.3220219.

[17] A. J. Feelders, "Credit scoring and reject inference with mixture models," *Intell. Syst. Accounting, Finance Manage.*, vol. 9, no. 1, pp. 1–8, Mar. 2000.

[18] Y. Liu, X. Li, and Z. Zhang, "A new approach in reject inference of using ensemble learning based on global semi-supervised framework," *Future Gener. Comput. Syst.*, vol. 109, pp. 382–391, Aug. 2020.

[19] R. Y. Goh and L. S. Lee, "Credit scoring: A review on support vector machines and metaheuristic approaches," *Adv. Oper. Res.*, vol. 2019, pp. 1–30, Mar. 2019.

[20] D. C. Hsia, "Credit scoring and the equal credit opportunity act," *Hastings LJ*, vol. 30, p. 371, Jan. 1978.

[21] J. Banasik, J. Crook, and L. Thomas, "Sample selection bias in credit scoring models," *J. Oper. Res. Soc.*, vol. 54, no. 8, pp. 822–832, Aug. 2003.

[22] H. T. Nguyen, "Reject inference in application scorecards: Evidence from France," EconomiX, Univ. Paris Nanterre, Nanterre, France, EconomiX Work. Papers 2016-10, 2016.

[23] J. Banasik and J. Crook, "Reject inference, augmentation, and sample selection," *Eur. J. Oper. Res.*, vol. 183, no. 3, pp. 1582–1594, Dec. 2007.

[24] M. Bücker, M. van Kampen, and W. Krämer, "Reject inference in consumer credit scoring with nonignorable missing data," *J. Banking Finance*, vol. 37, no. 3, pp. 1040–1045, Mar. 2013.

[25] S. Maldonado and G. Paredes, "A semi-supervised approach for reject inference in credit scoring using SVMs," in *Proc. ICDM*, 2010, pp. 558–571.

[26] Z. Li, Y. Tian, K. Li, F. Zhou, and W. Yang, "Reject inference in credit scoring using semi-supervised support vector machines," *Expert Syst. Appl.*, vol. 74, pp. 105–114, May 2017.

[27] Y. Tian, Z. Yong, and J. Luo, "A new approach for reject inference in credit scoring using kernel-free fuzzy quadratic surface support vector machines," *Appl. Soft Comput.*, vol. 73, pp. 96–105, Dec. 2018.

[28] R. A. Mancisidor, M. Kampffmeyer, K. Aas, and R. Jenssen, "Deep generative models for reject inference in credit scoring," *Knowl.-Based Syst.*, vol. 196, May 2020, Art. no. 105758.

[29] A. M. El, B. Benyacoub, and M. Ouzineb, "Semi-supervised adapted HMMs for P2P credit scoring systems with reject inference," *Comput. Statist.*, vol. 38, pp. 149–169, May 2022.

[30] Y. Kang, N. Jia, R. Cui, and J. Deng, "A graph-based semi-supervised reject inference framework considering imbalanced data distribution for consumer credit scoring," *Appl. Soft Comput.*, vol. 105, Jul. 2021, Art. no. 107259.

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.

[31] F. Shen, Z. Yang, X. Zhao, and D. Lan, "Reject inference in credit scoring using a three-way decision and safe semi-supervised support vector machine," *Inf. Sci.*, vol. 606, pp. 614–627, Aug. 2022.
[32] E. O. Ogundimu, "On lasso and adaptive lasso for non-random sample in credit scoring," *Stat. Model.*, May 2022, Art. no. 1471082X221092181.
[33] M.-L. Nguyen, T. Phung, D.-H. Ly, and H.-L. Truong, "Holistic explainability requirements for end-to-end machine learning in IoT cloud systems," in *Proc. IEEE 29th Int. Requirements Eng. Conf. Workshops (REW)*, Sep. 2021, pp. 188–194.
[34] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schlkopf, "Learning with local and global consistency," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2004, pp. 321–328.
[35] Y. Kang, R. Cui, J. Deng, and N. Jia, "A novel credit scoring framework for auto loan using an imbalanced-learning-based reject inference," in *Proc. IEEE Conf. Comput. Intell. Financial Eng. Econ. (CIFEr)*, May 2019, pp. 1–8.
[36] C. K. Chui and H. N. Mhaskar, "Deep nets for local manifold learning," *Frontiers Appl. Math. Statist.*, vol. 4, p. 12, May 2018.
[37] M. Kaya and H. S. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, p. 1066, Aug. 2019.
[38] K. Musgrave, S. Belongie, and S.-N. Lim, "A metric learning reality check," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 681–699.

Zhiyu Guo received the B.S. degree in information engineering from the Nanjing University of Information Science and Technology, Nanjing, China, in 2022. He is currently pursuing the M.S. degree with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing, China.

His research interests include graph representation learning and financial data mining.

Xiang Ao (Member, IEEE) received the B.S. degree in computer science from Zhejiang University, Hangzhou, China, in 2010, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China, in 2015.

He is an Associate Professor with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, CAS (ICT, CAS). His research interests include algorithms and models for AI finance tasks. He has authored more than 60 referred publications at prestigious international conferences and journals like IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, *ACM Transactions on Intelligent Systems and Technology*, ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), The Web Conference (WWW), IEEE International Conference on Data Engineering (ICDE), ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Annual Meeting of the Association for Computational Linguistics (ACL), AAAI Conference on Artificial Intelligence (AAAI), International Joint Conference on Artificial Intelligence (IJCAI).

Qing He (Member, IEEE) received the B.S. degree in mathematics from Hebei Normal University, Shijiazhuang, China, in 1985, the M.S. degree in mathematics from Zhengzhou University, Zhengzhou, China, in 1987, and the Ph.D. degree in fuzzy mathematics and artificial intelligence from Beijing Normal University, Beijing, China, in 2000.

He is a Professor with the Institute of Computing Technology, Chinese Academy of Science (CAS), Beijing, and he is also a Professor with the University of Chinese Academy of Sciences (UCAS), Beijing. His research interests include data mining, machine learning, classification, and fuzzy clustering.

Authorized licensed use limited to: Modern Education Society's Wadia College of Engineering. Downloaded on July 10,2024 at 09:32:21 UTC from IEEE Xplore. Restrictions apply.