# A DEEP LEARNING APPROACH FOR CREDIT SCORING

**Hoang Thanh Hai[1*], Than Quang Khoat[2]**
[1]*TNU - University of Economics and Business Administration*
[2]*Ha Noi University of Science and Technology*

**ARTICLE INFO**

**Received:** 17/01/2024
**Revised:** 14/5/2024
**Published:** 14/5/2024

**KEYWORDS**

Credit Scoring
Deep Learning
Profit
Data Imbalance
Data Shortage

**ABSTRACT**

Granting credit to customers is the core business of a bank. Hence, banks need adequate models to decide to whom to approve a loan. Over the past few years, the usage of deep learning to select appropriate customers has attracted considerable research attention. However, the data shortage, type of features, and data imbalance could decrease deep learning model performance from the accuracy perspective. This study aims to build a classifier for credit scoring based on deep learning. We use a credit scoring dataset publicly available on the UC Irvine Machine Learning Repository, a source of machine learning datasets commonly used by researchers. The model architecture is designed to be suitable for two kinds of input features, categorical and numerical ones. Our proposed model gave a relatively high accuracy among recent deep-learning-based models on the same dataset. We also consider the bank profit when applying the model, which is the ultimate goal of lenders. We found that if the banks use our model, they could gain a significant profit.

# MỘT MÔ HÌNH HỌC SÂU CHO BÀI TOÁN XẾP HẠNG TÍN DỤNG

**Hoàng Thanh Hải[1*], Thân Quang Khoát[2]**
[1]*Trường Đại học Kinh tế và Quản trị Kinh doanh - ĐH Thái Nguyên*
[2]*Đại học Bách khoa Hà Nội*

**THÔNG TIN BÀI BÁO**

**Ngày nhận bài:** 17/01/2024
**Ngày hoàn thiện:** 14/5/2024
**Ngày đăng:** 14/5/2024

**TỪ KHÓA**

Xếp hạng tín dụng
Học sâu
Lợi nhuận
Mất cân bằng dữ liệu
Thiếu hụt dữ liệu

**TÓM TẮT**

Cho vay tín dụng là hoạt động kinh doanh chủ yếu của một ngân hàng. Do đó, các ngân hàng cần một mô hình có độ chính xác cao để quyết định khách hàng nào được cho vay. Trong những năm gần đây, việc sử dụng học sâu để lựa chọn khách hàng phù hợp thu hút được sự quan tâm lớn. Tuy nhiên, việc thiếu hụt dữ liệu, sự đa dạng của loại dữ liệu, hay mất cân bằng trong dữ liệu có thể làm giảm độ chính xác của các mô hình phân loại dựa trên học sâu. Mục tiêu nghiên cứu của chúng tôi trong bài báo này là xây dựng một mô hình phân loại tín dụng dựa trên học sâu. Chúng tôi sử dụng bộ dữ liệu được công bố trên kho lưu trữ UC Irvine Machine Learning, một kho lưu trữ các bộ dữ liệu được sử dụng nhiều trong học máy. Kiến trúc mô hình được thiết kế để phù hợp với hai loại dữ liệu đầu vào của mô hình, dữ liệu định tính và dữ liệu định lượng. Mô hình được đề xuất có độ chính xác tương đối cao trong lớp các mô hình học sâu trên cùng bộ dữ liệu. Chúng tôi cũng xem xét lợi nhuận thu được của ngân hàng khi sử dụng mô hình. Kết quả cho thấy mô hình mang lợi mức lợi nhuận đáng kể cho ngân hàng.

**DOI: https://doi.org/10.34238/tnu-jst.9608**

[*] Corresponding author. *Email: hoangthanhhai03091988@gmail.com*

## 1. Introduction

There are two main approaches to assess credit risk: the judgmental approach and the statistical approach. A judgmental approach is a qualitative, expert-based approach whereby, based on business experience and common sense, the credit expert will make decisions about the credit risk [1]. The statistical one is a data-based method, where the lenders use historical data to find the relationship between a customer's characteristic and the binary (good or bad) target variable. This approach has many advantages compared with the judgmental one. It is better in terms of speed and accuracy. Moreover, it is also consistent. We no longer have to rely on the experience, intuition, or common sense of someone.

Recently, deep learning models have been increasingly used in credit scoring, besides classical machine learning models. This transition is partly thanks to great performances shown by deep learning in various real-world applications like image recognition, computer vision, or financial data analysis. However, the performance of DL-based classifiers does not really outperform that of classifiers without deep learning techniques in several credit scoring problems. This is partly due to the size and the type of input data. Hence, it is necessary to use suitable methods to deal with these problems. We proposed a data augmentation technique to generate more similar data to train model efficiently with small data. Credit data usually consist of both numeric and categorical features, so we need to build an appropriate model architecture that can learn two separate data types. Imbalance is also a common phenomenon with credit data, where customers classified as Bad are normally minor. In this paper, we combined some appropriate methods simultaneously in order to improve model performance using data encountered above mentioned issues. We then examined model improvement by the ablation study.

Credit scoring is a binary classification problem. The lenders want to build a classifier to divide the lenders into good and bad groups. Numerous statistical and machine learning methods have been applied in credit scoring over the years, such as logistic regression [2], neural networks [3], decision trees [4], or support vector machines [5]. Initially, the accuracy of these methods appeared to be limited. Recently, the performance of machine-learning-based models has improved considerably since the adoption of ensemble methods like bagging and boosting. Over the last years, the application of deep learning models in credit scoring has attracted considerable research attention. Up until now, various DL models have been applied. The most popular models are multilayer perceptrons (MLPs), convolution neural networks (CNNs), and long short-term memory (LSTM). LSTM networks are DL networks specifically designed for sequential data analysis. Wang et al. [6] used an attention mechanism LSTM network to predict the probability of user default in peer-to-peer lending. Dastile et al. [7] proposed converting tabular datasets into images and the application of CNNs in credit scoring. Li et al. [8] constructed a two-stage hybrid default discriminant credit scoring model based on deep forest. However, the performance of DL-based models is not always superior to classical machine learning classifiers, especially in case the data is small and includes various categorical features.

Making a profit is clearly the ultimate goal of any business. A credit scoring model that gives banks no profit could not be applied in actual business activity. Therefore, considering profit perspective when applying a classification model is crucial. Most studies consider the profit metric as an evaluation measure for the validation process rather than as an objective to be maximized in the training process of the model. These studies proposed some profit formula, then assessed the bank profit of the trained model. Profit formulas vary from source to source, depending on each author's assumptions or each bank's profit calculation.

The objectives of this study are twofold: (1) to conduct a classification model using a deep learning approach that gains relatively high accuracy although the dataset is small and imbalanced; (2) to consider the bank profit, whether it gains a profit or incurs a loss when using the model.

## 2. Methods

Our proposed framework consists of four stages. Firstly, we randomly split data into training and test sets. We use the training data to train the model and the test set to evaluate the final model. Credit datasets are often tabular data comprising various feature types, so we divide input columns into categorical and continuous ones. The initial training set is small, so we propose a method to augment training data. Oversampling is then applied with the training data to achieve equal split between two target classes. Next, input columns are fed into the network through separate layers, depending on their data category. Each type of column has its own linear layer to learn separately before concatenating into a shared four hidden layers network. The model architecture is shown in Figure 1. Then we train the overall network on the training set to achieve the best model. In the third stage, the final model is evaluated using different measurements. We do ablation study to assess the influence of each used method to the model performance. Finally, the bank profit is considered under some assumptions. The methodology undertaken in this study is discussed in detail in the following subsections.

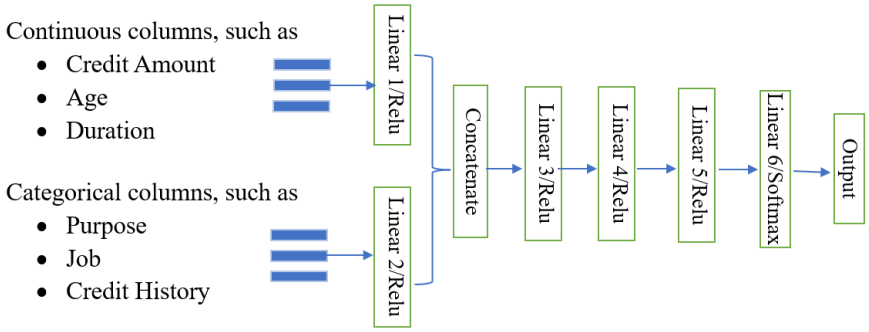


**Figure 1.** *The proposed model architecture*

### *2.1. Data augmentation*

Data augmentation is a technique of increasing the amount of data by generating new data instances from original data. It involves making minor changes to the data or using deep learning models to create new samples. In this study, we propose a data augmentation method by adding noise to instances. For each original data point, we create new samples by adding a tiny change amount to numerical features and remaining other categorical features unchanged. Let $D_{train} = \{(\boldsymbol{x}_i, y_i), i = 1, .., N\}$ be the training dataset, where $\boldsymbol{x}_i = (x_{i1}, \dots, x_{im}, z_{i1}, \dots, z_{in})$ are the features, $y_i$ is the credit status ($y_i = 1$ or $y_i = 0$) of the i[th] customer, $(x_{i1}, \dots, x_{im})$ are continuous and $(z_{i1}, \dots, z_{in})$ are categorical. Set

$$a_j = \max_{i=1,\dots,N} x_{ij}, j = 1, \dots, m.$$

The new continuous feature values are determined by

$$x'_{ij} = x_{ij} + r.u_{ij}, i = 1, \dots, N, j = 1, .., m$$

where $u_{ij}$ is sampled from uniform distributions $U(-a_j, a_j)$ and $r$ is a small parameter. The new samples corresponding to $(\boldsymbol{x}_i, y_i)$ have the following form

$$(x'_{i1}, \dots, x'_{im}, z_{i1}, \dots, z_{in}).$$

By sampling multiple times from above uniform distributions, we have a bigger training dataset including new sampled instances and the original ones.

### *2.2. Oversampling*

Imbalance is common in credit scoring datasets, where the great majority of data are good credit customers. Imbalanced data can cause deep learning models to favor the majority class while neglecting the minority class, leading to poor performance and biased predictions [9].

Random oversampling involves randomly duplicating examples from the minority class and adding them to the training class dataset. This can be repeated until the desired class distribution is achieved in the training dataset, such as an equal split across the classes. Applying a re-sampling strategy to obtain a more balanced dataset is considered an effective solution to the imbalance problem [10].

### *2.3. Weighted cross entropy loss*

Cross entropy is a commonly used loss function for classification problems. It is a measure of how different two distributions are. In credit scoring, we are interested in the difference between the true label and the predicted label of customers. Suppose we have a classification problem with two classes. Let $y = \{y_i = (y_{i1}, y_{i2}), i = 1, \dots, N\}$ be the true label distribution, where each element represents the probability of the corresponding class ($(0,1)$ or $(1,0)$). Let $p = \{p_i = (p_{i1}, p_{i2}), i = 1, \dots, N\}$ be the predicted label distribution. The cross entropy between $y$ and $p$ is given by

$$\frac{1}{N}\sum_{i=1}^{N}(-y_{i1}\log p_{i1} - y_{i2}\log p_{i2}).$$

The cross entropy is a non-negative value, with lower values indicating that the predicted distribution is closer to the true distribution.

In case we want the model to pay more attention to some class than others, we use weighted cross entropy. It is the cross entropy with each term multiplied by a weight factor. The weighted cross entropy between $y$ and $p$ in the binary classification is

$$\frac{1}{N}\sum_{i=1}^{N}(-w_1 y_{i1}\log p_{i1} - w_2 y_{i2}\log p_{i2}),$$

where $w = (w_1, w_2)$ is the weight vector with each element represents the weight for the corresponding class. The higher the weight, the more important the corresponding class is.

## 3. Experiments and Results

### *3.1. Data*

In this study, we use German, one of the most used datasets in credit scoring literature. It is publicly available on UC Irvine Machine Learning Repository [11]. The German dataset has 20 features of which 7 are numerical and 13 are categorical. These features include status of existing checking account, duration in month, credit history and purpose, to mention a few. The target variable is binary, i.e. customers are classified either as "Good" or "Bad". The imbalance ratio is 2.33 (700 – Good, 300 – Bad). The dataset has no missing values.

### *3.2. Model*

**Table 1.** *Parameters and Architecture of the model*

| Layer | Parameters and Architecture |
|---|---|
| Input | |
| Categorical Columns | Input shape: 13 |
| Continuous Columns | Input shape: 7 |
| Linear 1 | in_features = 7, out_features = 100 |
| Linear 2 | in_features = 13, out_features = 100 |
| Linear 3 | in_features = 200, out_features = 200 |
| Linear 4 | in_features = 200, out_features = 100 |
| Linear 5 | in_features = 100, out_features = 50 |
| Linear 6 | in_features = 50, out_features = 2 |

The dataset was split into 70% training set and 30% test set following the common practice in the literature [12]. We sampled four times from uniform distributions (with $r = 0.01$) to augment training data. The weight vector was $(10, 3)$, where 10 was the weight of Good class. The new training set has a size of 3500 (2450 – Good, 1050 – Bad) before oversampling process. The resampled dataset has a size of 4900 (2450 – Good, 2450 – Bad), which is the final dataset fed into the training process. Table 1 shows the parameters and architecture of the model.

The training set was used to determine the optimal weights and parameters of the best model. The test set was used to assess the performance of the final model. The model was trained using 200 epochs and a batch size of 128. All experiments for this study were performed in Python language using the Pytorch deep learning library. We used Adam method for optimizing parameters.

### *3.3. Results*

*Model Performance*

We assess the performance of our final model using the following metrics: Accuracy, Precision, Recall, and Area Under Receiver Operating Characteristic Curve (AUC). All metrics are measured after 10 random runs. Table 2 shows the model performance on the 30% test set. The model AUC is 0.764, which is considered to be good, as it indicates that the model has good discriminatory ability.

**Table 2.** *Model Performance (mean ± standard deviation)*

| Accuracy | Precision | Recall | AUC |
|---|---|---|---|
| $0.825 \pm 0.005$ | $0.843 \pm 0.011$ | $0.913 \pm 0.013$ | $0.764 \pm 0.010$ |

Table 3 shows the performance of our model and other related models in credit scoring for German dataset. Although our accuracy and AUC are lower than those of some models without deep learning techniques, they are relatively high amongst deep-learning-based classifiers. Recently, only two deep learning techniques have been proposed for the German dataset [13]. This is likely because classifying dataset with numerous categorical features is more challenging. With three used techniques (augmentation data, oversampling, weighted loss) and the two separate flow architecture, our model could be a reference model for those who have similar dataset type. Also, banks or credit institutions could consider using our model in their risk management system.

**Table 3.** *Performances of Related Models on German Credit Dataset*

| Study (Year) | Accuracy (%) | AUC | Methods |
|---|---|---|---|
| [14] (2021) | 79.5 | 0.831 | Without Deep learning techniques |
| [8] (2021) | 81.2 | **0.868** | Deep-learning-based |
| [15] (2009) | 82.0 | 0.824 | Without Deep learning techniques |
| [16] (2020) | 84.0 | 0.713 | Without Deep learning techniques |
| [17] (2018) | 85.78 | ----- | Without Deep learning techniques |
| [18] (2018) | 86.47 | ----- | Without Deep learning techniques |
| [7] (2021) | **88.0** | ----- | Deep-learning-based |
| [19] (2020) | 93.12 | ----- | Without Deep learning techniques |
| [20] (2021) | **98.66** | ----- | Without Deep learning techniques |
| **Ours** | **82.50** | **0.764** | Deep-learning-based |

### *3.4. Ablation study*

In this subsection, we do some ablation studies to investigate the influence of various components in our model on the model performance. We used three techniques, namely, augmentation data (AUG), oversampling (OVS), and weighted loss (WTL). We examine the impact of these techniques by removing them from the overall model. We also assess the effect of dividing input columns (DIV) into two separate flows before feeding into the network. The results are shown in Table 4. All experiments are performed over 10 random runs.

**Table 4.** *Ablation study using different component combinations (mean ± standard deviation)*

| AUG | OVS | WTL | DIV | Accuracy | Precision | Recall | AUC |
|---|---|---|---|---|---|---|---|
| | ✓ | ✓ | ✓ | 0.811 ± 0.007 | 0.831 ± 0.019 | 0.915 ± 0.035 | 0.741 ± 0.019 |
| ✓ | | ✓ | ✓ | 0.800 ± 0.008 | 0.807 ± 0.018 | **0.940** ± 0.023 | 0.706 ± 0.025 |
| ✓ | ✓ | | ✓ | 0.809 ± 0.006 | **0.871** ± 0.017 | 0.854 ± 0.023 | **0.778** ± 0.015 |
| | | | ✓ | 0.819 ± 0.003 | 0.857 ± 0.007 | 0.892 ± 0.010 | 0.772 ± 0.009 |
| ✓ | ✓ | ✓ | ✓ | **0.825** ± 0.005 | 0.843 ± 0.011 | 0.913 ± 0.013 | 0.764 ± 0.010 |
| ✓ | ✓ | ✓ | | 0.812 ± 0.004 | 0.824 ± 0.007 | 0.932 ± 0.013 | 0.733 ± 0.007 |
| | | | | 0.796 ± 0.037 | 0.812 ± 0.036 | 0.897 ± 0.059 | 0.704 ± 0.046 |

It is evident that each component in our model plays an important role and contributes to the final model performance. Oversampling has the most influence among three used techniques. When removing oversampling, model accuracy decreases significantly from 82.5% to 80%, and AUC falls sharply from 0.764 to 0.706. Model accuracy without data augmentation is 1.4% lower than full model, which means that data augmentation has a significant impact on model performance. A similar remark can be seen with weighted loss. Noticeably, splitting input columns into two separate flows do improve network performance. A classical multilayer perceptron (MLP) network with five hidden layers (i.e. our model without any techniques and without dividing input columns) has accuracy 79.6%, which is 2.3% lower than that of the model with two input flows. Overall, our proposed methods to deal with small, imbalanced tabular data have a beneficial effect on the final model.

### *3.5. Profit Consideration*

These performance metrics ultimately must be translated into profit consideration for the bank. Suppose that a correct decision of the bank would result in 25% profit. A correct decision here means that the lender predicts an application to be credit-worthy, and it turns out to be credit worthy. Conversely, when the bank classifies the application as good, but it turns out to be bad credit, the loss is 100%. Loan facility is not extended to applicants classified as non-creditworthy ones. Then the bank would not incur any loss [10].

In German dataset, 70 percent are creditworthy. Hence, a credit risk manager without any model would gain $0.7 \times 0.25 + 0.3 \times (-1) = -0.125$ profit, or incur 0.125 loss. The avarage loan amount is approximately 3270 DM, so the loss per applicant is about 408.75 DM.

Our model mean precision is 0.843. Therefore, the bank would gain

$$0.843 \times 0.25 + 0.157 \times (-1) \approx 0.054 \text{ profit.}$$

By using the proposed model, the bank would achieve $0.054 \times 3270 = 176.58$ DM profit per applicant, which is by far better than the result when the lender does not apply any models to manage credit risk.

## 4. Conclusion

This study applied a deep learning approach to the credit scoring problem. We trained the model using a multilayer perceptron network with two separate input flows and techniques like

data augmentation, oversampling, and weighted loss for imbalanced data on a small credit scoring dataset. Our proposed algorithm demonstrated promising results with relatively high accuracy and AUC amongst recent deep-learning-based models. The bank profit was also considered. By applying our model, the lender would gain a great profit per applicant. In future work, we focus on identifying methods and architectures to improve the performance of deep-learning-based credit risk models.

## REFERENCES


[1] B. Baesens, D. Rošch, and H. Scheule, *Credit risk analytics: measurement techniques, applications, and examples in SAS*, John Wiley & Sons, Inc, New Jersey, 2016.

[2] C. Serrano-Cinca and B. Gutiérrez-Nieto, "The use of profit scoring as an alternative to credit scoring systems in peer-to-peer (P2P) lending," *Decis. Support Syst.*, vol. 89, pp. 113–122, 2016.

[3] R. Setiono, B. Baesens, and C. Mues, "Recursive neural network rule extraction for data with mixed attributes," *IEEE Trans. Neural Netw*, vol. 19, pp. 299–307, 2008.

[4] J. R. Quinlan, *Programs for Machine Learning*, Morgan Kaufmann: San Mateo, Canada, 1993.

[5] D. Martens, B. Baesens, T. Van Gestel, and J. Vanthienen, "Comprehensible credit scoring models using support vector machines," *Eur. J. Oper. Res.*, vol. 183, pp. 1488–1497, 2007.

[6] C. Wang, D. Han, Q. Liu, and S. Luo, "A Deep Learning Approach for Credit Scoring of Peer-to-Peer Lending Using Attention Mechanism LSTM," *IEEE Access*, vol. 7, pp. 2161-2168, 2019.

[7] X. Dastile and T. Celik, "Making deep learning-based predictions for credit scoring explainable," *IEEE Access*, vol. 9, pp. 50426–50440, 2021.

[8] G. Li, H.D. Ma, R.Y. Liu, M. D. Shen, and K. X. Zhang, "A two-stage hybrid default discriminant model based on Deep Forest," *Entropy*, vol. 23, 2021, Art. no. 582.

[9] J. Nagidi, "Best ways to handle imbalanced data in machine learning," *Dataaspirant Homepage*, [Online]. Available: https://dataaspirant.com/handle-imbalanced-data-machine-learning/ [Accessed Jan. 15, 2024].

[10] J. Brownlee, "Random Oversampling and Undersampling for Imbalanced Classification," *Machine Learning Mastery Homepage*, [Online]. Available: https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/. [Accessed Jan. 15, 2024].

[11] German Credit Data, "UC Irvine Machine Learning Repository," [Online]. Available: https://archive.ics.uci.edu/dataset/144/statlog+german+credit+data. [Accessed Jan. 15, 2024].

[12] P. Selva, "Train test split – How to split data into train and test for validating machine learning models?" [Online]. Available: https://www.machinelearningplus.com/machine-learning/train-test-split/ [Accessed Feb. 23, 2024].

[13] H. Yoichi, "Emerging trends in deep learning for credit scoring: A review," *Electronics*, vol. 11, no. 19, 2022, Art. no. 3181.

[14] W. Zhang, D. Yang, S. Zhang, J. H. Ablanedo-Rosas, X. Wu, and Y. Lou, "A novel multi-stage ensemble model with enhanced outlier adaptation for credit scoring," *Expert Syst. Appl.*, vol. 165, 2021, Art. no. 113872.

[15] L. Yu, S. Wang, and K. K. Lai, "An intelligent-agent-based fuzzy group decision making model for financial multi criteria decision support: The case of credit scoring," *Eur. J. Oper. Res.*, vol. 195, pp. 942–959, 2009.

[16] N. Arora and P. D. Kaur, "A Bolasso based consistent feature selection enabled random forest classification algorithm: An application to credit risk assessment," *Appl. Soft Comput.*, vol. 86, 2020, Art. no. 105936.

[17] D. R. Edla, D. Tripathi, R. Cheruku, and V. Kuppili, "An efficient multi-layer ensemble framework with BPSOGSA-based feature selection for credit scoring data analysis," *Arab. J. Sci. Eng.*, vol. 43, pp. 6909–6928, 2018.

[18] D. Tripathi, D. R. Edla, and R. Cheruku, "Hybrid credit scoring model using neighborhood rough set and multi-layer ensemble classification," *J. Intell. Fuzzy Syst.*, vol. 34, pp. 1543–1549, 2018.

[19] S.K. Trivedi, "A study on credit scoring modeling with different feature selection and machine learning approaches," *Technol. Soc.*, vol. 63, 2020, Art. no. 101413.

[20] S. Acharya, I. V. Pustokhina, D.A. Pustokhin, B. T. Geetha, G. P. Joshi, J. Nebhen, E. Yang, and C. Seo, "An improved gradient boosting tree algorithm for financial risk management," *Knowl. Manag. Res. Pract.*, 2021, doi:10.1080/14778238.2021.1954489.