Knowledge Based Chinese Invoice Audit in Intelligent Financial Reimbursement System

Rui Xin, Xi Chen, Sisi Sun, Dan Jiang, and Bo Zhang(✉)

State Grid Hebei Information and Telecommunication Branch, Shijiazhuang, China
tony.htwang@foxmail.com

Abstract. Chinese invoice is one of the important types of commercial receipts for Chinese organizations. The audit of Chinese invoice guarantees the correctness and compliance for project management. To automatically finish the audit in intelligent financial reimbursement system, in this paper, we propose a knowledge based Chinese invoice audit framework. We firstly realize the invoice text recognition via the pp-ocrv3 model, and then classify the extracted text into commodities based on the BERT-TextCNN model. Finally, we construct a knowledge graph for rule-based reasoning to audit Chinese invoice. Our experiments and case study verified the effectiveness of the proposed framework.

Keywords: financial reimbursement system · Chinese invoice · knowledge graph · audit

1 Introduction

Traditional financial management is a human intensive task. The proportion of manual processing in financial work is high, and the level of intelligence is insufficient. Although financial reimbursement system has been built to ease the load, humans are also required to do many key tasks, e.g., invoice audit.

In recent years, 'Smart finance' came into being, breaking through the limitations of traditional financial management, using information technology means such as artificial intelligence to independently collect and analyze relevant financial information. Intelligent financial reimbursement system [1] is a highly desirable system which can provide knowledge for enterprises' accurate decision-making and commercial activities.

However, there are problems in financial reimbursement system, such as low processing efficiency, poor audit accuracy, and difficult data extraction. Among them, the audit of financial invoices requires business experts with certain financial knowledge to spend a lot of energy to complete the compliance judgment of more than thousands audit rules. Such complex and time-consuming business processes are easy to make auditors feel tired, and then induce errors and reduce audit efficiency. Therefore, it is necessary and important to reform and improve the invoice audit method.

In the digital era, intelligent invoice audit process has become necessary. Invoice recognition is the basis of invoice audit. The current invoice recognition function is generally implemented based on OCR (Optical Character Recognition). PP-OCR [2] is

N. Radojević et al. (Eds.): ICAID 2022, AHIS 7, pp. 28–37, 2023.
https://doi.org/10.2991/978-94-6463-010-7_5

a lightweight OCR system introduced by PaddleOCR. It considers the balance between accuracy and speed, and has strong practicability. The invoice audit function is to classify short text and construct knowledge graph for text data. Short text classification has been implemented based on BERT [3] or TextCNN [4]. The logic rules are formulated into the constructed knowledge graph and the invoice compliance can be checked on knowledge graph.

In this paper, we proposed a knowledge-based Chinese invoice audit framework in the intelligent financial reimbursement system. Specifically, we first use the pp-ocrv3 model to realize the invoice text recognition in the Chinese scene, then the extracted information is utilized to classify into commodities based on the BERT-TextCNN model. We next construct a knowledge graph and rules for financial reimbursement, and finally the compliance judgment of the invoice is realized via reasoning on the knowledge graph [5]. The experimental results show that our Chinese invoice audit framework can accurately and efficiently replace the manual invoice audit.

2 Proposed Framework

2.1 System Overview

The system realizes the Chinese invoice compliance monitoring in intelligent financial reimbursement system based on the knowledge graph. The system architecture is shown in Fig. 1. The input is generally an image. Firstly, the invoice recognition is realized through OCR, and the image format data is transformed into the processable text structured data. Based on the above structured data, the Chinese short text classifier extracts the features of commodity description information and realizes commodity classification. Further, based on the structured invoice data and Chinese short text classification results, the knowledge graph is constructed to complete invoice compliance audit.

2.2 Invoice Recognition

The invoice recognition function in Chinese scene is realized by using the PP-OCRv3 model realized by the paddle framework. The system receives an invoice image as input and then segment the invoice image region based on the regularity of the invoice to further recognize the region of interest in invoice. The detection module is used to detect the text line in segmented regions. A text direction classifier is added to classify and rectify the text detection boxes. The recognition module recognizes the contents of the processed text box. Finally, Merge information from all recognized regions and output a structured data.

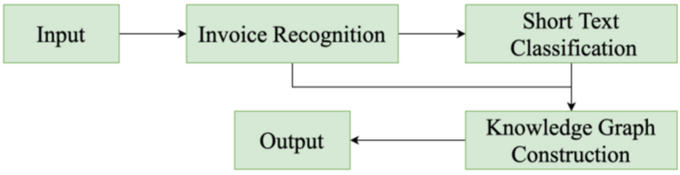

Fig. 1. The Proposed Framework

1) *Text detection:* Text detection is to locate the position of text in the image. The detector is optimized based on Differentiable Binarization (DB) [6], which is a text detection algorithm based on segmentation. The DB can better deal with irregular shaped text such as bending and simplify the post-processing process of segmentation.
2) *Detection Boxes Rectify:* Rectify detection boxes is to rect the detected text line in the non-zero degree image, providing a horizontal rectangular box for subsequent recognition. The rectification includes two stages: classification and geometric transformation. A text direction classifier is a simple image classifier trained to classify text from a specified angle. Affine transformation acts on text detection box with non-zero angle to realize rectification.
3) *Text Recognition:* Text recognition is based on the text box obtained by the detection model to further recognize the text in the text box. The recognition module is optimized based on STVR [7], which is a single visual model for scene text recognition. The SVTR replaces the RNN structure with transformers structure to mine the context information of text line image more effectively.

2.3 Short Text Classification

In this section, we first introduce the mathematical definition of text classification task. Next, we will provide an introduction to our BERT-TextCNN model.

1) ***Problem Formulation***: Given a corpus of m texts $X = \{x_1, x_2, ..., x_m\}$, $x_i \in \mathbb{R}^n$, and a corresponding set of m labels $Y = \{y_1, y_2, ..., y_m\}$, $y_i \in \mathbb{R}$. Text classifier attempts to train a model $F : \mathcal{X} \rightarrow \mathcal{Y}$, $\mathcal{X}$ and $\mathcal{Y}$ denote the input and the output space respectively. Specifically, suppose there is an input text $x \subseteq \mathcal{X}$, the classifier can give a predicted true label y_{true} based on a posterior probability P.

$$\underset{y_i \in \mathcal{Y}}{\operatorname{argmax}} P(y_i|x) = y_{true} \tag{1}$$

This paper is based on the BERT and TextCNN network for text classification, the frame is as shown in Fig. 2. First, it is necessary to train the classification model, that is, fine-tune BERT to improve the effect of BERT in downstream tasks, and then classify the commodities to be reimbursed based on the trained classification model.
2) ***Structure of BERT***: BERT is a Mask Language Model (MLM) [8] for pre-training and uses Deep Bidirectional Transformer [9] to build the whole model. Therefore, a deep bidirectional language representation that can integrate left and right context information is finally generated.

BERT only uses the encoder part of Transformer. The input of encoder is a two-dimensional matrix X, and each row represents a sentence. The Embedding layer represents each word in the sentence as a vector, and outputs a three-dimensional matrix $X_{embedding}$, which respectively represents the number of sentences in a batch, the number of words in each sentence, and the embedding dimension of each word. Then the positional Encoding is used to mark the position information of each word, an encoding

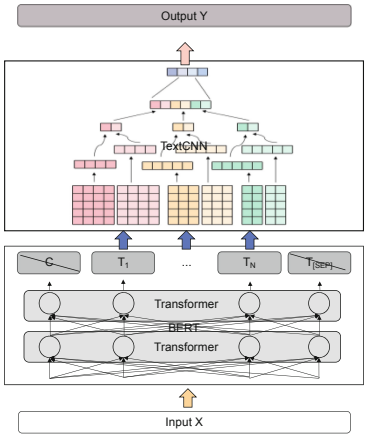

Fig. 2. Structure of BERT-TextCNN Model (Fig. 2 is original and does not need permission)

matrix X_{pos} that is the same as the input dimension can be obtained, and add it on the original token embedding to get a new embedding by:

$$X_{embedding} = X_{pos} + X_{embedding} \tag{2}$$

The core of the Encoder is the Attention module, which utilize the attention mechanism to output the enhanced semantic vector representation of words. The calculation steps of Attention are as follows:

- Calculate the query matrices Q, the queried matrices K and the actual feature matrices V, which are obtained through model training:

$$\begin{aligned} Q &= X_{embedding} \times W^Q \\ K &= X_{embedding} \times W^K \\ V &= X_{embedding} \times W^V \end{aligned} \tag{3}$$

- Calculate attention:

$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where $Softmax()$ is a normalized exponential function and d_k is the dimension of the word vector. The self-attention mechanism in BERT is a special case of the attention mechanism, namely $Q = K = V$.

Unlike Transformer, BERT uses Multi-head-Self-Attention mechanism to calculate the attention value by:

$$\begin{aligned} &MultiHeadAttention(Q, K, V) \\ &= Concat(head_1, head_2, \ldots, head_k)W^o \\ &head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right) \end{aligned} \tag{4}$$

where W_i^Q, W_i^K and W_i^V represent the weight matrix of the i-th head for W^Q, W^K and W^V respectively, W^o measures the additional weight matrix, and $Concat()$ is the concatenation function. The output of the Encoder will go through an Add&Norm layer. Add means adding the input and output of the Multi-head-Self-Attention layer, and Norm means that the added output will be normalized so that it has a fixed mean and standard deviation, where the mean is 0 and the standard deviation is 1. Then the normalized vector list will be passed to a fully connected feedforward neural network. Similarly, the FeedForward layer will also be processed by the corresponding Add&Norm layer, and output the normalized hidden layer vector X_{hidden}.

3) ***Structure of TextCNN:*** Convolutional Neural Network (CNN) is used for graph processing, as its variant model Text Convolutional Neural Network (TextCNN) extracts local features of different sizes in text sequences by setting filtering kernels of different sizes. The word vector matrix is used as the input of TextCNN, and the whole process of TextCNN is divided into embedding layer, convolution layer, pooling layer, fusion layer and fully connected layer.

The most important module in the TextCNN model is the convolution layer, which requires fewer parameters than other deep learning models, and convolution can extract different features of the input information. The convolutional layer is composed of several convolution kernel modules, the TextCNN model in this paper contains 3 convolution kernels of size (2, 3, 4). The convolution kernel can also be called a filter, and its related parameters are optimized by the back-propagation algorithm. The filter has a set of neurons with fixed weights. Therefore, the filter considers the word order relationship between adjacent words, and after the calculation is completed, a column vector representing the feature extracted by the filter from the sentence can be obtained. Each filter can get a feature, so the convolution layer can extract local features of the text, that is, get different feature expressions. The calculation formula is as follows:

$$a_i = f(W \times T_{i:i+n-1} + b) \tag{5}$$

where W is the weight, b is the bias, n is the size of the convolution kernel, f is a nonlinear function, and $T_{i:i+n-1}$ represents the word vector at different positions in the text. The obtained a_i is the i-th feature extracted by a convolution kernel. The convolution kernel width of the convolutional layer of Text-CNN is fixed, and its size is equal to the dimension of the word vector. The convolution kernel generates features respectively according to the sliding order and then obtains the feature vector.

After going through the convolutional layer, the obtained features are often of large dimension and need to be reduced in dimension. The role of the pooling layer is to

reduce the dimensionality of the input and prevent over-fitting, the main method is to use abstract analysis to reduce the number of parameters to a certain extent. The fusion layer splices and fuses the features obtained by the pooling layer into a vector, but it is still a local feature. At this time, a fully connected layer is required to combine all local features into global features. The fully connected layer adds a hidden layer after the fusion layer, and finally classify short text through the Softmax function.

2.4 KG Construction and Reasoning

In this subsection, we extract key knowledge concepts for the financial reimbursement system and reimbursement process. The ontology design [10] of constructed financial reimbursement knowledge graph and specific knowledge examples are shown in Fig. 3. The part above the dotted line in the figure is the conceptual template of financial reimbursement knowledge. The part below the dotted line is a specific instance that Alice is reimbursed for a $1,299 electronics product for a Science Foundation project.

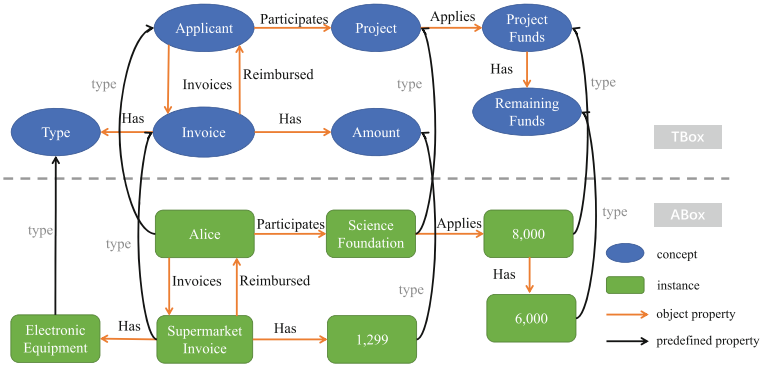

Fig. 3. The Ontology of Financial Reimbursement Knowledge Graph

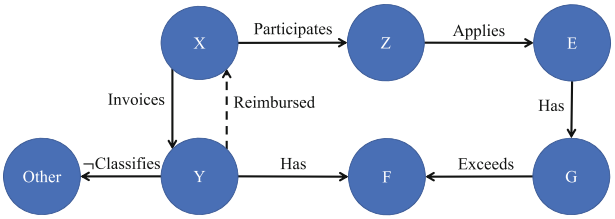

Fig. 4. The Logic Rules

Using the constructed knowledge graph and the formulated logic rules, simple logical reasoning for reimbursement can be realized. As shown in Fig. 4, the logic rules based on the financial reimbursement system are as follows:

$$\begin{aligned} &Invoices(X, Y) \wedge Participates(X, Z) \wedge \\ &Applies(Z, E) \wedge \neg Classifies(Y, Other) \wedge \\ &Has(Y, F) \wedge Has(E, G) \wedge Exceeds(G, F) \\ &\qquad \rightarrow Reimbursed(Y, X) \end{aligned} \tag{6}$$

The formulate idea is to satisfy that the reimbursement amount declared by the applicant shall not exceed the remaining funds, and the reimbursement type must be compliant.

3 Experiments

3.1 Invoice Recognition

The example of invoice recognition is shown in Fig. 5. The figure shows an instance of recognizing the commodity area in the invoice image. Segment the commodity area based on the regularity of the invoice and recognize the segmented image respectively by OCR. The figure also shows the segmentation result of the product name and the recognition result after segmentation. Finally, the recognition results are combined into a structured dictionary for downstream short text classification tasks.

3.2 Results of Short Text Classification

We evaluate the performance of the proposed classification framework experimentally in this section. We firstly present the experiment setup, and then report the experiment results on the preprocessed dataset. The results show that our framework could achieve much better performance in short text classification.

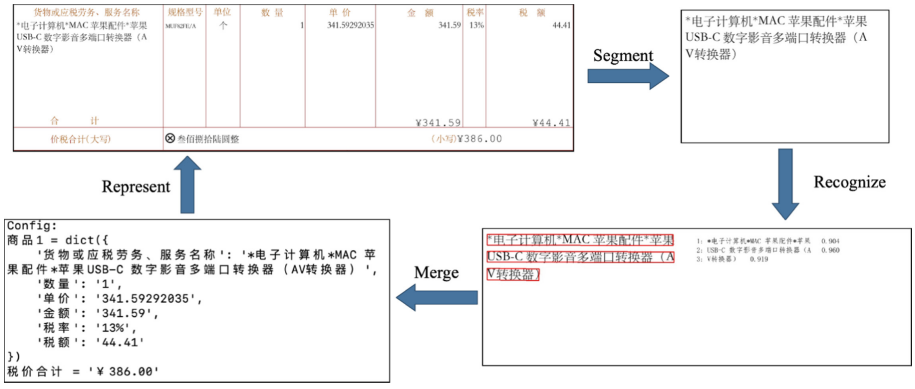

Fig. 5. Example of Invoice Recognition

1) ***Dataset:*** The experimental data is taken from the 10th College Student Service Outsourcing Competition-2018 Online Retail Platform Commodity Classification Data Set (CCDS), including 50w commodities and corresponding 1296 category labels. However, there are non-standard data in this data. After data preprocessing, a total of 12760 pieces of data are finally obtained for experiments, details of the datasets are listed in Table 1. Train, Dev, and Test represent the number of samples in the training, validation, and test datasets. Class denotes the classification number. Avg Len means the average sentence length. We finally integrate these data into four categories according to the categories of common reimbursement products: electronic equipment, office supplies, books and others.
2) ***Base Models:*** We used five main classic deep neural networks as base models in our framework for short text classification task: RNN, LSTM, Bi-LSTM, TextCNN, and BERT.
3) ***Evaluation:*** We evenly sampled each class from the origin data to form 360 testing examples for every datasets. Then these test samples are used to examine the effectiveness of the classification model. The key metrics to evaluate the performance of different models in this paper is Accuracy, which refers to the ratio that the number of correctly predicted examples against the total number of testing examples, and the better the model, the higher the accuracy rate.

   Table 2 shows the accuracy results of base models and our model for short text classification. For each base model, we highlight the highest classification accuracy, and it can be seen that our model performs the best classification.
4) ***Practical Application:*** We identify the invoices in the real world, extract the identified "commodity name" field to get the commodity text information, manually label its category, and finally get 113 pieces of real-world test set *Yreality*. Then, we use our trained classification model and classify it to verify the effect of our classification

Table 1. Detailed information of datasets.

Dataset Name	Train	Dev	Test	Class	Avg Len
CCDS	12000	400	360	4	30

Table 2. The evaluation results under different settings.

Model Name	Accuracy%
RNN	96.39
LSTM	96.94
Bi-LSTM	94.44
TextCNN	97.78
BERT	98.01
BERT − TextCNN (ours)	**98.30**

Table 3. Instances of knowledge reasoning.

Knowledge	Answer
Invoices (Wang, Supermarket Invoice), Participates (Wang, Science Foundation), Applies (Science Foundation, 8000), Classifies (Supermarket Invoice, Books), Has (Supermarket Invoice, 299), Has (8000, 6000) Exceeds (6000, 299)	Reimbursed (Supermarket Invoice, Wang)
Invoices (Li, Supermarket Invoice), Participates (Li, Science Foundation), Applies (Science Foundation, 1,000), Classifies (Supermarket Invoice, Electronic Equipment), Has (Supermarket Invoice, 1,200) Has (1000, 800)	Reimbursed (Supermarket Invoice, Li)

model. The final classification accuracy is **96.43%**, which proves that our work is of practical significance and can effectively classify the goods to be reimbursed in the invoice.

3.3 Case Study

We show two instances of knowledge reasoning in Table 3. The first example is when Wang applied for reimbursement of $299 for books for a Science Foundation project. Since the declared amount is less than the remaining funds, and the reimbursement type is not other, it satisfies the logic rules model to classify texts in the invoice into commodities, and a knowledge graph with logic rules for reasoning on the financial properties of commodities. Our experiments and case study demonstrated the effectiveness of the proposed framework on the task of Chinese invoice audit.

The reasoning result is reimbursement. The second instance is when Li applied for reimbursement of $1200 for electronic equipment for a Science Foundation project. It does not meet the conditions of *Exceeds*() because the declared amount exceeds the remaining funds. The reimbursement request is rejected.

4 Conclusion

In this paper, we propose a framework for the audit of Chinese invoice in financial reimbursement system. Our framework includes an OCR module to recognize texts from Chinese invoice via pp-ocrv3 model, a short text classification model to classify texts in the invoice into commodities, and a knowledge graph with logic rules for reasoning on the financial properties of commodities. Our experiments and case study demonstrated the effectiveness of the proposed framework on the task of Chinese invoice audit.

Acknowledgment. This paper is supported by the Science and Technology Project of State Grid Hebei Information and Telecommunication Branch: The Function Improvement of Intelligent Financial System Based on Artificial Intelligence Technology (No. SGHEXT00SJJS2100251).

References

1. Lan X (2021) Intelligent early warning method for financial sharing center expense reimbursement audit. In: Proceedings of the 3rd international conference on artificial intelligence and advanced manufacture, 23–25 October 2021, Manchester, United Kingdom, pp 1569–1572
2. Du Y et al. (2020) PP-OCR: a practical ultra lightweight OCR system. arXiv preprint arXiv:2009.09941
3. Devlin J, Chang M, Lee K, Toutanova K (2019) BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics, NAACL-HLT 2019, 2–7 June 2019, Minneapolis, MN, USA
4. Kim Y (2014) Convolutional neural networks for sentence classification. CoRR
5. Hu P, Urbani J, Motik B, Horrocks I (2019) Datalog reasoning over compressed RDF knowledge bases. In: Proceedings of the 28th ACM international conference on information and knowledge management, CIKM 2019, 3–7 November 2019, Beijing, China, pp 2065–2068
6. Liao M, Wan Z, Yao C, Chen K, Bai X (2020) Real-time scene text detection with differentiable binarization. In: Proceedings of the AAAI conference on artificial intelligence, vol 34, no 07, pp 11474–11481
7. Du Y, et al (2022) SVTR: Scene text recognition with a single visual model. arXiv preprint arXiv:2205.00159
8. Ghazvininejad M, Levy O, Liu Y, Zettlemoyer L (2019) Mask-predict: parallel decoding of conditional masked language models. In: Proceedings of the 2019 conference on empirical methods in natural language processing, EMNLP-IJCNLP 2019, 3–7 November 2019, Hong Kong, China, pp 6111–6120
9. Vaswani A et al (2017) Attention is all you need. In: Proceedings of the annual conference on neural information processing systems 2017, Long Beach, CA, USA, 4–9 December 2017, pp 5998–6008
10. Liu JNK, He Y, Lim EHY, Wang X (2013) A new method for knowledge and information management domain ontology graph model. IEEE Trans Syst Man Cybern Syst 43(1):115–127