STATISTICAL ANALYSIS AND DATA MINING

An In-domain Pre-training and Prompt Learning-Based Data Analysis Method for the Steel E-commerce Industry

Journal:	*Statistical Analysis and Data Mining*
Manuscript ID	SAM-23-308
Wiley - Manuscript type:	Research Article
Date Submitted by the Author:	15-Aug-2023
Complete List of Authors:	Peng, Qiaojuan; University of Science and Technology Beijing School of Computer and Communication Engineering; University of Science and Technology Beijing Shunde Innovation School; Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing Key Laboratory of Knowledge Engineering for Materials Science Luo, Xiong; University of Science and Technology Beijing School of Computer and Communication Engineering; University of Science and Technology Beijing Shunde Innovation School; Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing Key Laboratory of Knowledge Engineering for Materials Science Yuan, Yuqi; University of Science and Technology Beijing School of Computer and Communication Engineering; University of Science and Technology Beijing Shunde Innovation School; Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing Key Laboratory of Knowledge Engineering for Materials Science Gu, Fengbo; Ouyeel Co Ltd Shen, Hailun; Ouyeel Co Ltd Huang, Ziyang; Ouyeel Co Ltd
Keywords:	Data analysis, Text classification, In-domain pre-training, Prompt learning

Note: The following files were submitted by the author for peer review, but cannot be converted to PDF. You must view these files (e.g. movies) online.

manuscript.bbl
sn-mathphys.bst

An In-domain Pre-training and Prompt Learning-Based Data Analysis Method for the Steel E-commerce Industry

Qiaojuan Peng[1,2,3], Xiong Luo[1,2,3*], Yuqi Yuan[1,2,3], Fengbo Gu[4], Hailun Shen[4], Ziyang Huang[4]

[1]School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China.
[2]Shunde Innovation School, University of Science and Technology Beijing, Foshan 528399, China.
[3]Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing 100083, China.
[4]Ouyeel Co., Ltd., Shanghai 201999, China.
ORCID: 0009-0008-8667-1468 (Qiaojuan Peng).

*Corresponding author(s). E-mail(s): xluo@ustb.edu.cn;
Contributing authors: d202210397@xs.ustb.edu.cn;
M202110668@xs.ustb.edu.cn; gufengbo@ouyeel.com;
shenhailun@ouyeel.com; huangziyang@ouyeel.com;

Abstract

With the continuous growth of business, a large number of quality objection texts have been accumulated on steel e-commerce platforms. These quality objection texts usually express consumer dissatisfaction with the dimensions, specifications, appearance, and performance of steel products, which have valuable insights for the improvement of steel products and consumer decision-making. In this context, precise classification of quality objection texts has become a pressing issue that requires immediate attention. Currently, the mainstream solution is to use pre-trained models. However, there are two main drawbacks to these models: 1) their performance on domain-specific datasets is not as good as on general-domain datasets; 2) their performance on few-shot datasets is not satisfactory. To address these two issues, this paper presents an advanced data analysis method on the basis of **i**n-domain **p**re-training, **b**idirectional **e**ncoder **r**epresentation from

Transformers, and **p**rompt **l**earning (IP-BERT-PL). Specifically, 1) a domain-specific unsupervised dataset is introduced into the BERT model for in-domain pre-training, enabling the model to better understand specific language patterns and features in the steel e-commerce industry, enhancing the model's generalization capability; 2) the idea of prompt learning is incorporated into the BERT model, allowing the model to pay more attention to contextual information of sentences, improving the model's classification performance on few-shot datasets. By comparing with mainstream methods through experimental evaluation, our method has demonstrated superior performance on the quality objection dataset. Additionally, the introduction of prompt learning has not only enhanced the model's performance but also accelerated its computation speed, surpassing the original BERT model. Furthermore, the ablation experiments further validate the significant advantages of in-domain pre-training and prompt learning in improving model performance, while also showcasing the outstanding performance of prompt learning on few-shot datasets.

Keywords: Data analysis, Text classification, In-domain pre-training, Prompt learning

1 Introduction

With the popularity of e-commerce, people are increasingly inclined to shop and trade online [1, 2]. In the steel e-commerce industry, with the rapid development of steel e-commerce, steel transaction has shifted from traditional offline markets to online platforms. Consumers can easily purchase various specifications and varieties of steel products through e-commerce platforms. As the scale of steel transaction continues to expand, consumers' attention to the quality and performance of steel is also increasing, and expressing dissatisfaction with the quality of steel has become more and more frequent. These quality objection texts usually express consumers' dissatisfaction with the dimensions, specifications, appearance, performance, and other aspects of steel products. By accurately classifying quality objection texts, steel manufacturers can gain valuable insights into specific issues or defects that impact their products. For example, they can identify common problems such as surface irregularities, dimensional inconsistencies, or material defects. With this information, manufacturers can make informed decisions to improve their manufacturing processes, optimize product design, and address quality issues promptly. Additionally, precise classification of quality objection texts directly influences consumer decision-making. Steel products are widely used in various industries, including construction, automotive, and infrastructure. When consumers consider purchasing steel products, they rely on information about the quality and performance of different steel options. Accurate classification of objection texts provides consumers with reliable data and insights, enabling them to make informed choices based on their specific requirements. In this context, how to accurately identify and classify quality objection texts has become a pressing issue that requires immediate attention.

Traditional text classification methods for quality objection texts in the steel e-commerce industry rely on manual classification. This method has two main drawbacks: 1) subjectivity: manual text classification is heavily influenced by subjective factors, and different people may have different results when classifying the same text, leading to unstable classification results; 2) high workload: with the continuous expansion of steel transaction scale, the accumulated quality objection texts are increasing, and manually classifying a considerable number of quality objection texts demands significant resources, time, and workforce, resulting in high costs.

As a data analysis technique, text classification aims to automatically analyze and process text to classify it into pre-defined categories. Text classification has wide applications in many fields such as information retrieval, sentiment analysis, and news classification. It can help extract useful information from massive amounts of text, quickly locate the required text, and speed up information processing. Therefore, we can use text classification technology to automatically classify quality objection texts in the steel e-commerce industry.

Currently, existing text classification methods can be broadly categorized into rule-based methods, machine learning-based methods, and deep learning-based methods. Rule-based methods rely on manually designed rules to classify texts using prior knowledge and experience. This method has the advantage of being fast and easy to adjust, but it requires domain experts to design rules, making it difficult to adapt to changes in data and new situations. Machine learning-based methods utilize statistical learning to train a classifier from the training data and then utilize it to classify new data. Common machine learning algorithms include support vector machines (SVM) [3], random forest [4], decision tree [5], and others. This method is advantageous in terms of simple algorithm implementation, ease of understanding and adjustment, and good performance on small-scale datasets. Deep learning-based methods employ deep neural networks (DNNs) to model and learn from text data, allowing for more complex and sophisticated representations of text. Typical deep learning models include recurrent neural network (RNN) [6], convolutional neural network (CNN) [7], Transformer [8], and bidirectional encoder representations from Transformers (BERT) [9]. The main benefit of this method is that it can use large-scale datasets for training and automatically learn effective feature representations, resulting in good performance on complex text classification tasks. As a consequence, deep learning-based methods have gained prominence as the primary approach for text classification tasks, particularly with the emergence of pre-trained models like BERT, which has become a popular solution in the natural language processing (NLP) field. BERT is trained on massive data and learns more general and comprehensive language representations, which improves its generalization ability. Compared with previous methods, BERT adopts two pre-trained tasks, masked language model (MLM) and next sentence prediction (NSP), enabling the model to learn deeper language rules and contextual relationships, thereby improving its performance in downstream tasks. The rise of pre-trained models has brought new ideas and methods to the NLP field, allowing people to better utilize large-scale data and powerful computing capabilities to solve practical problems.

However, current mainstream pre-trained models, although performing well in the field of NLP, still have certain limitations. First, the parameters of these pre-trained

models are trained on general corpora. If the training corpora used for pre-training differ from those used for downstream tasks, the model parameters obtained from pre-training may not be well transferred to downstream tasks, which reduces the model's transferability. Second, such pre-trained models require fine-tuning on specific task datasets. If the training set is small in size, the performance of the model may diminish.

As this paper focuses on quality objection data in the steel e-commerce industry, which has strong domain-specific characteristics, the general corpus used in pre-trained models differs significantly from the steel corpus. In addition, there is limited annotated data in the steel e-commerce industry. Therefore, these two limitations are particularly evident in the dataset used in this paper. To address these two issues, this paper proposes an advanced data analysis method based on in-domain pre-training, BERT, and prompt learning (IP-BERT-PL). Specifically, to enhance the transferability of the model, a domain-specific corpus is incorporated into the BERT model for in-domain pre-training. Furthermore, to enhance the performance of the BERT model on few-shot dataset, prompt learning is integrated, owing to the limited availability of training data.

In summary, this paper's main contributions are:

1. By incorporating domain-specific unsupervised corpora into the in-domain pre-training process of the BERT, the model gains a better understanding of the language patterns and features specific to the steel e-commerce industry, thereby improving its ability to generalize in that domain.
2. By integrating the concept of prompt learning into the BERT, traditional text classification tasks are transformed into cloze-style tasks, enabling the model to focus more on the contextual information of sentences and thereby enhancing its classification performance.
3. Our model performs best in the quality objection dataset, and the ablation experiments further demonstrate the superiority of our proposed model.

The other sections of this paper are arranged as follows. Section 2 mainly introduces and compares some methods for implementing text classification tasks. Section 3 provides a detailed description of the text classification model proposed in this paper. Section 4 compares the implementation effects of different models in text classification on the quality objection dataset in the steel e-commerce industry, and carries out the ablation experiments to demonstrate the advantages of the method proposed in this paper. Finally, a summary of the work in this paper is presented.

2 Related Work

As discussed in Section 1, existing text classification methods can be broadly categorized into rule-based methods, machine learning-based methods, and deep learning-based methods. This section mainly focuses on these three categories of text classification methods for detailed introduction.

2.1 Rule-based Methods

The rule-based method refers to the approach of text classification using manually defined rules. This method usually requires expertise in a particular field, annotated data, and feature extraction and selection. In rule-based method, it is typically necessary to define some rules first, such as keywords, vocabulary, syntax structures, etc., and assign weights to these rules. Then the text is fed into a classifier that determines the text's category based on the assigned weights of the rules.

Han *et al.* introduced a context-dependent word clustering method based on rules [10]. Their experiments showed a considerable reduction in dimensionality and an 8% enhancement in the overall accuracy of bibliographic field extraction from references. Thabtah *et al.* used four different rule-based methods to classify the CCA Arabic text [11]. The study indicated that the text classification method based on OneRule demonstrated the poorest performance on the CCA dataset, whereas the method based on C4.5 exhibited the highest performance on the same dataset.

In summary, rule-based methods have some advantages, such as the ability to ensure correct classification of specific cases by defining rules, and may be more effective for small datasets than other methods. However, there are also some drawbacks to this method, such as the need for significant time and effort to manually define rules, and the potential for rules to become outdated due to changes in the data. With the evolution of machine learning techniques, the scope of application for rule-based methods is becoming increasingly limited.

2.2 Machine Learning-based Methods

The machine learning-based method is a way to classify text by training machine learning algorithms. It learns from a large amount of annotated data to automatically extract features and patterns in the text, achieving automatic text classification. The method usually requires feature extraction and selection for text, such as bag-of-words model [12], term frequency–inverse document frequency (TF-IDF) [13], etc.

Harrag *et al.* utilized decision trees to classify Arabic text documents and conducted experiments on two datasets [5]. The experimental outcomes demonstrated that the decision tree-based method was effective in achieving good results on both datasets. Shah *et al.* employed logistic regression, random forest, and KNN to classify news texts from BBC [14]. The authors tested, analyzed, and compared these three classifiers. The experimental outcomes showed that the classifier based on logistic regression had the highest accuracy on the dataset, reaching 97%. The random forest classifier achieved the second-highest accuracy of 93%, while KNN had the lowest accuracy of 92% among the three algorithms. Nevertheless, all three classifiers yielded satisfactory results.

In summary, machine learning-based methods are advantageous for their fast training speed and high classification accuracy, and are suitable for medium-sized datasets. However, this method also has some drawbacks, such as difficulty in handling large-scale datasets and high-dimensional feature spaces, as well as the need for manual

feature selection and optimization of algorithm parameters. As deep learning technology continues to advance, traditional machine learning methods are gradually being replaced.

2.3 Deep Learning-based Methods

The deep learning-based method use DNNs to classify text. Its significant difference from the machine learning-based method is that the latter often requires manual feature extraction and poses a challenge in dealing with the semantic content of the text, while the former can automatically learn features from raw text, and is an end-to-end method, thus improving the accuracy of text classification.

Lai *et al.* utilized a RNN to learn word representations with contextual information, which is more effective in reducing noise than traditional window-based neural networks [15]. Additionally, a max-pooling layer was integrated to identify key features in the text, which was essential for text classification. The researchers carried out experiments on four datasets, and the findings indicated the superiority of their proposed approach, especially on document-level datasets. Sun *et al.* conducted a comprehensive study to explore various fine-tuning methods of BERT for text classification tasks [16]. The study demonstrated that BERT achieved the best performance compared to other models on eight commonly used text classification datasets.

Given the impressive performance of pre-trained models like BERT in various NLP domains, an increasing number of scholars tend to utilize BERT for implementing downstream tasks. However, as mentioned in Section 1, there are two main issues with current pre-trained models: 1) insufficient transferability on domain-specific datasets, and 2) poor performance on few-shot datasets. Therefore, the proposed IP-BERT-PL model aims to introduce in-domain pre-training, and prompt learning to respectively address the above-mentioned two issues.

3 Methodology

This section mainly introduces the IP-BERT-PL method in detail.

3.1 Task Description

Given a set of sentences $X = \{x_1, x_2, \cdots, x_i, \cdots, x_m\}$ and a set of labels $C = \{c_1, c_2, \cdots, c_j, \cdots, c_n\}$, the objective of our model is to find the corresponding label c_j for each sentence x_i. Among them, m denotes the number of sentences, and n represents the number of labels.

3.2 The Pre-trained Model of BERT

Based on a Transformer encoder, BERT is trained bidirectionally on a large corpus and learns rich language representations. BERT's training consists of two stages: pre-training and fine-tuning. In the pre-training stage, BERT uses a bidirectional encoder to train the model. Unlike traditional unidirectional language models, BERT considers contextual information and introduces two tasks: MLM and NSP. In the MLM task, some words in the input sentence are replaced with a special token "[MASK]",

and the model is then tasked with predicting these masked words. This task's training process can help the model better understand lexical semantics and syntactic structure, and perform context reasoning, thereby improving the model's representational capacity at the word level. In the NSP task, BERT takes two sentences as input and determines whether they are semantically related, that is, whether these two sentences are adjacent. This task's training process can help the model better understand the relationship and contextual semantics between sentences, thereby improving the model's representational capacity at the sentence level. In the fine-tuning stage, BERT adapts the pre-trained model to perform specific tasks, such as text classification and named entity recognition. During fine-tuning, BERT's bidirectional pre-training can better capture contextual information in tasks, achieving breakthrough performance improvements in various NLP tasks.

3.3 Model Architecture

Fig. 1 shows the architecture of the IP-BERT-PL model. The proposed model consists of two stages: in-domain pre-training and fine-tuning. In this paper, we first construct an unsupervised dataset specific to the steel e-commerce industry and perform incremental pre-training of the BERT model on this dataset, known as in-domain pre-training. This process enables the BERT model to better understand the textual content in the steel e-commerce domain and capture semantic relationships within it. Then, based on the concept of prompt learning, we transform the original quality objection dataset into a corpus containing prompt templates and fine-tune the BERT model. This enables BERT to generalize to the task of quality objection text classification and improves its performance and accuracy on this specific task. The following will provide a detailed introduction to these two stages of in-domain pre-training and fine-tuning.

3.3.1 In-domain Pre-training

Because our dataset consists of quality objection texts in the steel e-commerce industry, it has a very obvious domain specificity. To enhance the model performance on the quality objection dataset, in-domain pre-training is required. In this paper, in-domain pre-training based on the BERT model refers to the process of pre-training on unsupervised steel industry corpora to provide domain-relevant semantic understanding and representation capabilities. It mainly involves the following four steps.

Step 1: Construct a dataset for in-domain pre-training. Firstly, obtain raw quality objection texts submitted by customers from the steel e-commerce platform. Then, the texts undergo data cleaning to prepare for in-domain pre-training. The main steps of data cleaning include removing duplicate texts as they do not provide additional information for model training. Additionally, texts unrelated to quality objections are removed to ensure that the dataset only contains content relevant to quality issues. Furthermore, redundant punctuation marks are eliminated to reduce noise and improve the learning effectiveness of the model. Finally, the cleaned texts are merged into an unsupervised dataset $X = \{x_1, x_2, \cdots, x_i, \cdots, x_m\}$, where x_i represents the i-th text and m denotes the length of dataset X.

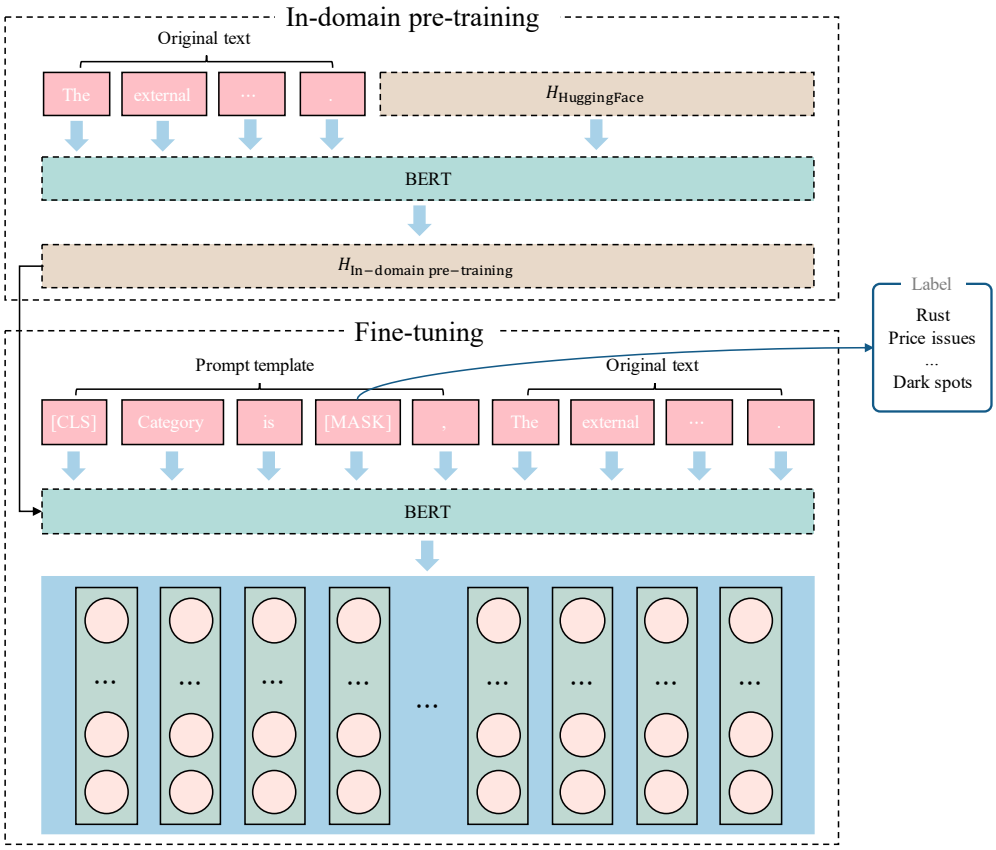


Fig. 1 The architecture of the IP-BERT-PL model

Step 2: Load the open-source Chinese pre-trained model parameter package, $H_{\text{HuggingFace}}$[1] from HuggingFace. To initialize the BERT model, it is necessary to load the pre-trained model parameters provided by HuggingFace. This pre-trained model has been trained on a large-scale Chinese corpus and possesses rich language knowledge and representation capabilities. By utilizing this parameter package, the model can be initialized to a state with basic language understanding abilities, thus providing a solid starting point for subsequent in-domain pre-training tasks.

Step 3: Conduct in-domain pre-training for MLM task. From Section 3.2, it can be inferred that the BERT model undergoes two main tasks during the pre-training process: MLM and NSP. Since this paper focuses on quality objection text, which typically consists of single sentences rather than multiple sentences, and many experiments have indicated that the NSP task doesn't significantly contribute to model accuracy improvement [17, 18], the in-domain pre-training in this paper only involves training the MLM task. During the in-domain pre-training process, the in-domain pre-training

[1] https://huggingface.co/bert-base-chinese

dataset X and the Chinese pre-trained parameter package $H_{\text{HuggingFace}}$ are input into the MLM task of the BERT model for training. The MLM task aims to achieve true contextual joint modeling, similar to fill-in-the-blanks questions. For each sentence x_i in the dataset X, MLM randomly masks 15% of the tokens in the text and replaces them with the special token "[MASK]". The Transformer model is then used to predict the masked tokens and reconstruct them to their original forms. Approximately 15% of the tokens are selected, with 80% being replaced with "[MASK]", 10% being replaced with random tokens, and the remaining 10% being left unchanged. Through this masked-reconstruction process, the model can learn the semantic information of the masked tokens and their contextual relationships, enabling word-level semantic modeling.

Step 4: Calculate the MLM loss and update model parameters. The formula for calculating $Loss_{\text{MLM}}$ is given by (1). After training, we obtain the in-domain pre-training model parameter package, $H_{\text{In-domain pre-training}}$, which includes the model parameters trained specifically in the given domain. This parameter package can be used as the initialization parameters for subsequent tasks, providing better domain adaptability and performance.

$$Loss_{\text{MLM}} = -\sum_{i=1}^{M} \log p\left(m = m_i \mid \theta, \theta_1\right), m_i \in [1, 2, \cdots, |V|], \tag{1}$$

where θ corresponds to the parameters of the encoder in the BERT model, while θ_1 refers to the parameters of the output layer connected to the encoder for the MLM task. The set of masked words is denoted as M, $|V|$ represents the vocabulary size of BERT.

To summarize, by following the aforementioned steps, we have successfully accomplished the in-domain pre-training task, which allows the model to learn semantic representations of quality objection data at word-level. As a result, the performance of the model will be enhanced to some extent.

3.3.2 Fine-tuning

In the fine-tuning stage, we utilize the weights of the in-domain pre-trained model as the initial weights and train the model using labeled data specific to the domain, aiming to adapt the model to the specific requirements of that domain. In this process, we introduce the concept of prompt learning to assist the model in better understanding the language structure and semantic information relevant to the specific task. Below, we provide a detailed description of the steps involved in the fine-tuning stage:

Step 1: Firstly, we select a portion of data from the in-domain pre-training dataset X and have domain experts from the steel e-commerce industry annotate these data, forming a labeled dataset $T = \{(t_1, y_1), (t_2, y_2), \cdots, (t_i, y_i), \cdots, (t_n, y_n)\}$, where t_i represents the i-th text in the dataset T, and y_i represents the label corresponding to t_i, n is the length of the dataset T.

Step 2: Based on the idea of prompt learning, the labeled dataset T is transformed into a labeled dataset P that includes prompt templates. Prompt learning is a technique that utilizes language models to directly model the probability of text.

Specifically, the original input t_i is transformed into a modified string prompt t_i' by applying a template that contains unfilled slots. The language model then probabilistically fills in the missing information, resulting in a final string. The output y_i can be derived from this final string. The process involves two steps. First, a template is used, consisting of a text string with two slots: $[X]$ for the input t_i and $[Z]$ for an intermediate answer text z_i, which will be mapped to the final output y_i. Second, the input slot $[X]$ is filled with the input text t_i. For instance, in news text classification, if the input text t_i is "The basketball game ended with a score of 111:62.", a template like "This is a $[Z]$ news, $[X]$" can be applied. Applying this template, the modified input t_i' becomes "This is a $[Z]$ news, The basketball game ended with a score of 111:62.". Considering the quality objection text classification task in the steel e-commerce industry addressed in this paper, which consists of a total of 21 categories, precautions are taken to prevent the model from developing biases towards specific categories due to the prompt templates. The prompt templates designed in this paper are relatively objective and neutral, as shown in (2). Subsequently, we transform the labeled dataset T into a labeled dataset $P = \{(t_1', y_1), (t_2', y_2), \cdots, (t_i', y_i), \cdots, (t_n', y_n)\}$ that includes the prompt templates based on (2).

$$PT = \text{类别是}[Z], [X](\text{ in Chinese}), PT = \text{ Category is}[Z], [X](\text{in English}). \quad (2)$$

In the case of implementing this operation using the BERT model, the label prediction slot $[Z]$ is actually represented by the "[MASK]" token. Therefore, the subsequent $[Z]$ in the text is represented as [MASK].

Step 3: Determine the mapping between the predicted words and the true labels, i.e., determine the mapping between z_i and y_i. Since this paper focuses on text classification tasks where the label categories are predefined, we can directly map the predicted z_i to y_i. Generally, one [MASK] token represents the prediction of one word. In English text classification tasks, we can use one token to map each label. However, in Chinese text classification tasks, multiple tokens are often required. For example, in (3), we can use just one [MASK] token to predict the label word "Sports" in a classification task. But in (4), we need to use two [MASK] tokens to map the label word "体育" (which has the same meaning as "Sports"), and their probability product is used to represent the label probability. The quality objection dataset in this paper contains a total of 21 labels, and the lengths of these labels vary, ranging from 2 to 8. This data characteristic results in a varying number of [MASK] tokens, and as the number of [MASK] tokens increase, it becomes difficult for the model to predict all of them correctly. To address this issue, we adopt a simple and convenient solution, which is to add these 21 labels as special symbols to the vocabulary of the BERT model. Suppose the original length of the BERT vocabulary is V, the modified vocabulary length would be V+21. With this approach, we can use just one [MASK] token to map each label.

t_i' =This is a [MASK] news, The basketball game ended with a score of 111: 62. (3)

t_i' =这是一条 [MASK][MASK]新闻, 篮球比赛的比分是111: 62。 (4)

Step 4: The labeled dataset P containing prompt templates is divided into training, validation, and testing sets. Initially, the dataset is split into a training set and a testing set following a 4:1 ratio, which is a common approach to ensure an adequate amount of data for model training and evaluation. Subsequently, 20% of the training set is further partitioned to create a validation set. This validation set serves as an independent subset to evaluate and optimize the model's performance during the training process. After these steps, the dataset P is divided into training set: validation set: testing set in a ratio of 16:4:5. This division strategy aims to strike a balance between having a sufficiently large training set for effective model learning and utilizing a validation set to monitor and optimize the model's performance.

Step 5: Load the model parameter package $H_{\text{In-domain pre-training}}$ obtained during the in-domain pre-training phase and train the BERT model using the training set obtained in Step 4. Through iterative training, the model will adjust based on the specific task label information in the dataset to meet the requirements of the target task.

Step 6: During the fine-tuning process, we evaluate the performance of the model using the cross-entropy loss function. Specifically, we compare the model's predicted category for the masked tokens with the corresponding true labels at their respective positions and calculate the cross-entropy loss between them. This allows us to measure the model's prediction accuracy and error on the task-specific data.

In summary, through the above steps, we can obtain a fine-tuned model. During the fine-tuning stage, by training with task-specific data, the model can learn relevant features for the specific task, resulting in better classification performance. The introduction of prompt learning can help the model better learn the language structure and grammar rules specific to a particular domain, further improving the model's performance.

4 Experiments

In this section, we will provide a detailed description of the experimental setup and related details.

4.1 Dataset

4.1.1 In-domain Pre-training Dataset

The in-domain pre-training dataset used in this paper comes from customer complaints about the quality of steel products on a steel e-commerce platform. Since the original quality objection contents contain a lot of noise, we conduct data cleaning on them. The specific data cleaning operations can be found in Step 1 of Section 3.3.1. Finally, we obtain 16,500 quality objection texts related to steel products, which serve as the in-domain pre-training dataset.

4.1.2 Fine-tuning Dataset

Steel industry experts select 5425 data from the in-domain pre-training dataset, and annotate and verify them repeatedly, resulting in a fine-tuning dataset (quality objection dataset) covering 21 categories. Fig. 2 shows the distribution of labels in the quality objection dataset.

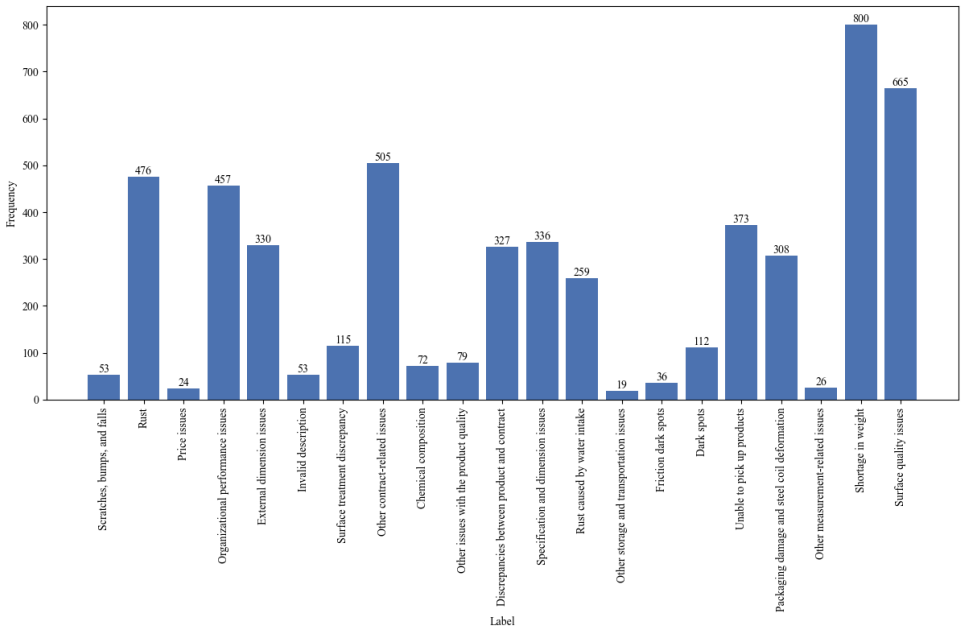


Fig. 2 The label distribution of the quality objection dataset

4.2 Evaluation Metric

In text classification tasks, accuracy is the most intuitive evaluation metric, representing the proportion of correctly predicted samples by the classifier out of the total samples. However, in imbalanced datasets, accuracy can be misleading as it fails to distinguish between misclassifications of different categories. As shown in Fig. 2, the dataset in this paper exhibits label imbalance. Therefore, the evaluation metrics used to assess the performance of the text classification model in this paper include precision, recall, and $F1$-score. Their definitions are as follows.

Precision: The proportion of correctly predicted positive samples out of all samples predicted as positive by the model, as represented by (5).

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{5}$$

Recall: The proportion of correctly predicted positive samples out of all actual positive samples, as represented in (6).

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{6}$$

where the definitions of TP, FP, and FN in (5) and (6) are as follows.

- TP (True Positive): It represents the cases where the model correctly classifies positive samples as positive.
- FP (False Positive): It represents the cases where the model incorrectly classifies negative samples as positive.
- FN (False Negative): It represents the cases where the model incorrectly classifies positive samples as negative.

$F1$-score: It is a balanced measure of the model's performance, which is calculated as the harmonic mean of precision and recall. It mainly evaluates the model's ability to correctly identify both positive and negative samples, and is only high when both precision and recall are relatively high. Thus, this paper primarily focuses on the $F1$-score metric, which can be calculated using (7).

$$F1\text{-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{7}$$

To evaluate the experimental results, we adopt a comprehensive approach that takes into account the arithmetic mean of each label. Specifically, we calculate precision, recall, and $F1$-score separately for each label in the dataset. Then, we average the precision, recall, and $F1$-score across all labels to obtain macro-precision, macro-recall, and macro-$F1$. These metrics provide an assessment of the model's overall performance on all labels, highlighting its effectiveness in handling various categories. They are suitable for assessing the model's performance in imbalanced datasets.

4.3 Parameter Settings

Table 1 shows the parameter settings for the in-domain pre-training and fine-tuning stages of the proposed method. The meaning of each parameter is as follows:

- batch_size: Used to control the number of samples input to the model in each training step, which affects the training speed and performance of the model.
- learning_rate: It is utilized to regulate the pace of updating the model parameters in each iteration of training, which in turn determines the convergence of the objective function to a local minimum and the optimal time to converge.
- num_epochs: Used to specify the number of times to traverse the entire training set during model training. In-domain pre-training stage, in order to prevent the model from overfitting, we add an early stopping mechanism [19].
- max_seq_length: It is set to define the maximum number of tokens that can be included in the input sequence. According to statistics, in our dataset, 99.94% of the data length is within 128. Therefore, we set max_seq_length here to 128.

Furthermore, all experiments in this paper are performed under Python 3.7.15 using Torch 1.10.2 and Transformers 4.18.0 libraries. Meanwhile, the optimizer used during the training stage is Adam.

Table 1 Parameter settings in the stage of in-domain pre-training and fine-tuning

Stage	batch_size	learning_rate	num_epochs	max_seq_length
In-domain pre-training	32	-	500	128
Fine-tuning	32	3e-5	5	128

4.4 Baseline Methods

To evaluate the performance of the proposed method in this paper, we compare it with several baseline methods. The descriptions of these baseline methods are provided below.

TextCNN [7]: It is a CNN that utilizes one-dimensional convolutional layers to extract features from text and selects the most prominent features through max pooling operations.

TextRCNN [15]: Refer to Section 2.3.

LSTM [20]: It is a variant of RNN that captures and manages long-term dependencies in sequence data by utilizing memory cells and gating units.

Bi-LSTM [6]: It is a bidirectional long short term memory (Bi-LSTM) network that combines the advantages of CNN and RNN. It can capture more comprehensive contextual information and is more effective in handling long sequences.

Att-BLSTM [21]: It is an extension of the traditional Bi-LSTM that introduces an attention mechanism to enhance the focus on key information from different positions in the sequence.

FastText [22]: It segments the text into word or character-level n-grams and treats these n-grams as features. This enables a better capture of the internal structure and semantic information of the words.

DPCNN [23]: It constructs a deep pyramid structure by employing multiple layers of convolution and pooling operations to model both local and global information of the text.

Transformer [8]: It utilizes self-attention mechanism to model the dependency relationships between different positions in a sequence, thereby capturing the global contextual information.

BERT [9]: Refer to Section 3.2.

4.5 Comparison with Other Methods

In Table 2, we present the performance comparison results of different methods on the quality objection dataset. From the experimental results, it can be observed that the IP-BERT-PL model performs the best on the quality objection dataset. This indicates that by introducing in-domain pre-training and prompt learning, we enable the model to better understand the semantics and context of the quality objection dataset,

thereby improving the model's classification performance. Compared to the BERT model, the IP-BERT-PL model's Macro-$F1$ score has increased by 1.62%, demonstrating the significant potential and value of the proposed method in handling the quality objection dataset. Additionally, this experimental result also suggests that for specific domain datasets, introducing in-domain pre-training and prompt learning methods may lead to better performance, providing useful inspiration for related research.

Table 2 Performance comparison of different methods on the quality objection dataset

Method	Macro-precision	Macro-recall	Macro-	Prediction time
TextCNN	90.96%	86.29%	87.37%	0.04s
TextRCNN	92.05%	88.48%	90.01%	0.14s
LSTM	79.61%	79.26%	78.36%	0.10s
Bi-LSTM	87.51%	80.19%	82.70%	0.20s
Att-BLSTM	90.01%	87.32%	88.43%	0.20s
FastText	87.79%	78.39%	80.84%	0.03s
DPCNN	84.69%	81.34%	82.72%	0.07s
Transformer	84.61%	77.67%	79.56%	0.12s
BERT	93.00%	91.13%	91.70%	3.51s
IP-BERT-PL	**94.36%**	**92.72%**	**93.32%**	**3.11s**

Since this study focuses on the steel e-commerce domain, practical application scenarios prioritize model performance and prediction time. Therefore, in Table 2, we also note the prediction time for each model on the test set (a total of 1,108 texts) in addition to recording the model performance. It can be observed that although the BERT-based models have a noticeable increase in prediction time compared to traditional machine learning algorithms, they also show a significant improvement in performance. We consider that real-time requirements are not essential in the steel e-commerce domain, and performance takes precedence, making this time overhead acceptable. Furthermore, we observe that our proposed model not only improves performance compared to the BERT model but also reduces the prediction time. This is because, in traditional classification tasks, the model needs to understand and encode the entire input and make predictions for each category, resulting in higher computational costs. However, by introducing the idea of prompt learning, we transform the classification task into a cloze-style task. In the cloze-style task, each token has a fixed position, and the model only needs to select the correct token to fill in based on the context. This constraint makes the prediction more direct and straightforward, reducing the computational complexity and improving the computational speed. Therefore, we conclude that our method demonstrates its effectiveness from both the perspectives of model performance and computational speed.

4.6 Ablation Studies

To further demonstrate the effectiveness of the proposed method, this section will conduct ablation experiments on the key components of the IP-BERT-PL method.

4.6.1 Effect of In-domain Pre-training

Table 3 presents the ablation experiments on in-domain pre-training, which plays a crucial role in our research. The experimental results provide strong evidence demonstrating the advantage of our paper's contribution as a necessary component for a domain-specific model. Specifically, our model achieves a significant improvement of 1.13% in the Macro-$F1$ score compared to the BERT model, thanks to the in-domain pre-training. These findings emphasize the importance of utilizing in-domain pre-training to enhance the language understanding capability of models in the steel e-commerce industry context. By learning industry-specific knowledge such as language rules, domain-specific terminology, and specific semantic relationships, our model becomes more suitable for handling tasks in this domain, thereby improving the performance of text classification tasks in this field.

Table 3 The effect of in-domain pre-training

Method	Macro-Precision	Macro-Recall	Macro-$F1$
BERT	93.00%	91.13%	91.70%
IP-BERT	94.28%	91.73%	92.83%

Note: The IP-BERT model involves incorporating in-domain pre-training into the BERT architecture.

4.6.2 Effect of Prompt Learning

Table 4 shows the results of ablation experiments on prompt learning. We can see that the Macro-$F1$ of the BERT-PL model with prompt learning outperforms the BERT model by 1.22%. This indicates that introducing prompt learning can provide the model with some prior knowledge, thereby improving the model's generalization ability. Moreover, since relevant prompt information is introduced to the model during training, the model is more likely to learn the correlation and semantic relationships between labels, thereby enhancing the model's ability to handle label sparsity problems. The experimental results in Table 4 also indicate that introducing prompt learning is necessary for improving the performance of the model on specific domains and label sparse datasets.

Table 4 The effect of prompt learning

Method	Macro-Precision	Macro-Recall	Macro-$F1$
BERT	93.00%	91.13%	91.70%
BERT-PL	94.75%	91.52%	92.92%

Note: The BERT-PL model involves incorporating prompt learning into the BERT architecture.

In addition, to investigate the impact of individual prompt templates on the model's performance, Table 5 presents the performance of the BERT-PL model using various prompt templates. From the results in Table 5, we can find that different

prompt templates have a huge impact on model performance. For example, for prompt templates "标签是[MASK]，[X]" and "类别是[MASK]，[X]", which differ by only one word, the Macro-$F1$ differs by 2.87%. These experimental results also indicate that the current models are extremely sensitive to prompt templates, and small changes can cause significant disturbances in model results. At the same time, not all prompt templates have a positive effect on improving the model's performance, and some may even reduce the effectiveness of the model. In general, although prompt learning can provide some prior knowledge to the model to help it better adapt to unseen data and improve its generalization ability, these prompt templates need to be manually designed, and designing a suitable prompt template demands significant time and effort. Therefore, in the future, it is of great research value to explore ways to reduce the model's sensitivity to prompt templates and to design an automatic prompt template construction method.

Table 5 Comparison of model performance with different prompt templates

Prompt (in Chinese)	Prompt (in English)	Macro-Precision	Macro-Recall	Macro-$F1$
标签：[MASK]，	Label: [MASK],	94.73%	90.40%	91.86%
类别：[MASK]，	Category: [MASK],	92.93%	89.67%	90.73%
分类：[MASK]，	Classification: [MASK],	92.31%	89.90%	90.73%
类型：[MASK]，	Type: [MASK],	93.92%	89.10%	90.86%
标签是[MASK]，	Label is [MASK],	91.93%	89.06%	90.05%
类别是[MASK]，	Category is [MASK],	94.75%	91.52%	92.92%
分类是[MASK]，	Classification is [MASK],	91.81%	91.12%	91.29%
类型是[MASK]，	Type is [MASK],	94.47%	91.60%	92.84%
钢材发生[MASK]，	Steel has [MASK],	93.92%	91.66%	92.41%
钢卷发生[MASK]，	Steel coil has [MASK],	93.48%	91.23%	92.05%
钢材发生：[MASK]，	Steel has: [MASK],	90.67%	89.31%	89.66%
钢卷发生：[MASK]，	Steel coil has: [MASK],	92.33%	91.65%	91.66%
质量异议的标签是[MASK]，	Label for quality objection is [MASK],	93.53%	90.42%	91.60%
质量异议的类别是[MASK]，	Category for quality objection is [MASK],	93.78%	92.02%	92.61%
质量异议的分类是[MASK]，	Classification for quality objection is [MASK],	93.70%	89.54%	91.23%
质量异议的类型是[MASK]，	Type for quality objection is [MASK],	92.58%	90.99%	91.43%
申诉说明的标签是[MASK]，	Label for objection description is [MASK],	92.60%	88.94%	90.38%
申诉说明的类别是[MASK]，	Category for objection description is [MASK],	93.59%	90.61%	91.73%
申诉说明的分类是[MASK]，	Classification for objection description is [MASK],	93.64%	90.69%	91.86%
申诉说明的类型是[MASK]，	Type for objection description is [MASK],	91.75%	90.25%	90.64%

Note: All results are without in-domain pre-training, and represent the output generated by the BERT-PL model.

Table 6 Comparison of model performance under few-shot scenario

Method	Macro-Precision	Macro-Recall	Macro-$F1$
BERT	88.95%	88.27%	88.35%
IP-BERT	90.53%	89.27%	89.54%

Furthermore, one major advantage of incorporating prompt learning is its applicability to few-shot scenarios. To demonstrate this advantage, we conduct corresponding experiments on the quality objection dataset. According to the standard document released by the China Electronics Standardization Association[2], for text classification tasks, a data volume of fewer than 200 instances per category is considered as few-shot. Following this criterion, we initially create a few-shot dataset. Specifically, we will randomly remove categories in the quality objection dataset that have more than 200 instances until there are only 200 instances remaining. At the same time, we will keep all the data with label quantities equal to or less than 200 instances. Subsequently, we perform comparative experiments using this few-shot dataset. Table 6 presents the impact of introducing prompt learning on the model's performance in the few-shot dataset. The results demonstrate that with the introduction of prompt learning, IP-BERT achieves a 1.19% improvement in the Macro-$F1$ metric compared to the original BERT model. This strongly proves that incorporating prompt learning can significantly enhance the performance of few-shot datasets and validates the necessity of introducing prompt learning in the quality objection dataset.

5 Conclusion

In this paper, we propose a novel IP-BERT-PL model for the classification of quality objection texts in the steel e-commerce industry. By incorporating domain-specific corpora for in-domain pre-training in the general pre-trained BERT model, we gain a better understanding of language patterns and features specific to the steel e-commerce industry, thereby improving the model's classification performance. Additionally, by introducing the idea of prompt learning into the BERT model, we successfully transform the traditional text classification task into a cloze-style task, allowing the model to focus more on contextual information and enhancing its generalization ability for quality objection text classification in the steel e-commerce industry. Experimental results demonstrate that the IP-BERT-PL model outperforms other mainstream methods in handling text classification tasks in the steel e-commerce industry. Furthermore, the ablation experiments further validate the significant advantages of in-domain pre-training and prompt learning in improving model performance, while also showcasing the outstanding performance of prompt learning on few-shot datasets.

In future, we will continue to explore how to further optimize the IP-BERT-PL model to meet the needs of different application scenarios. For example, as mentioned in Section 4.6.2, the current model is sensitive to prompt templates, and even a slight difference in prompt templates can cause drastic fluctuations in the model's final performance. Therefore, it is of great research value to explore ways to reduce the

[2] https://www.ttbz.org.cn/StandardManage/Detail/28143/

model's sensitivity to prompt templates and to design an automatic prompt template construction method.

Acknowledgments. This work was supported in part by the Beijing Natural Science Foundation under Grants L211020 and M21032, in part by the National Natural Science Foundation of China under Grants U1836106, 62271045, and U2133218, and in part by the Scientific and Technological Innovation Foundation of Foshan under Grants BK21BF001 and BK20BF010.

Conflicts of Interest. The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Fedushko, S., Ustyianovych, T.: E-commerce customers behavior research using cohort analysis: A case study of covid-19. Journal of Open Innovation: Technology, Market, and Complexity **8**(1), 12 (2022)

[2] Aliyev, A.G.: Problems of regulation and prospective development of e-commerce systems in the post-coronavirus era. Information Engineering and Electronic Business **14**(6), 14–26 (2022)

[3] Hearst, M.A., Dumais, S.T., Osuna, E., Platt, J., Scholkopf, B.: Support vector machines. IEEE Intelligent Systems and Their Applications **13**(4), 18–28 (1998)

[4] Breiman, L.: Random forests. Machine Learning **45**, 5–32 (2001)

[5] Harrag, F., El-Qawasmeh, E., Pichappan, P.: Improving arabic text categorization using decision trees. In: Proceedings of the 1st International Conference on Networked Digital Technologies, Ostrava, Czech Republic, pp. 110–115 (2009). IEEE

[6] Liu, P., Qiu, X., Huang, X.: Recurrent neural network for text classification with multi-task learning. In: Proceedings of the 25th International Joint Conference on Artificial Intelligence, pp. 2873–2879. IJCAI/AAAI Press, New York, NY, USA (2016)

[7] Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1746–1751. ACL, Doha, Qatar (2014)

[8] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems, Long Beach, CA, USA, pp. 5998–6008 (2017)

[9] Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics:

Human Language Technologies, pp. 4171–4186. Association for Computational Linguistics, Minneapolis, MN, USA (2019)

[10] Han, H., Manavoglu, E., Giles, C.L., Zha, H.: Rule-based word clustering for text classification. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 445–446. ACM, Toronto, Canada (2003)

[11] Thabtah, F., Gharaibeh, O., Abdeljaber, H.: Comparison of rule based classification techniques for the arabic textual data. In: Proceedings of the 4th International Symposium on Innovations in Information and Communications Technology, Amman, Jordan, pp. 105–111 (2011). IEEE

[12] Zhang, Y., Jin, R., Zhou, Z.: Understanding bag-of-words model: a statistical framework. International Journal of Machine Learning and Cybernetics **1**, 43–52 (2010)

[13] Hakim, A.A., Erwin, A., Eng, K.I., Galinium, M., Muliady, W.: Automated document classification for news article in bahasa indonesia based on term frequency inverse document frequency (TF-IDF) approach. In: Proceedings of the 6th International Conference on Information Technology and Electrical Engineering, Yogyakarta, Indonesia, pp. 1–4 (2014). IEEE

[14] Shah, K., Patel, H., Sanghvi, D., Shah, M.: A comparative analysis of logistic regression, random forest and KNN models for the text classification. Augmented Human Research **5**(12), 1–16 (2020)

[15] Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: Proceedings of the 29th AAAI Conference on Artificial Intelligence, pp. 2267–2273. AAAI Press, Austin, USA (2015)

[16] Sun, C., Qiu, X., Xu, Y., Huang, X.: How to fine-tune BERT for text classification? In: Proceedings of the 18th China National Conference on Chinese Computational Linguistics, vol. 11856, pp. 194–206. Springer, Kunming, China (2019)

[17] Joshi, M., Chen, D., Liu, Y., Weld, D.S., Zettlemoyer, L., Levy, O.: Spanbert: Improving pre-training by representing and predicting spans. Transactions of the Association for Computational Linguistics **8**, 64–77 (2020)

[18] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)

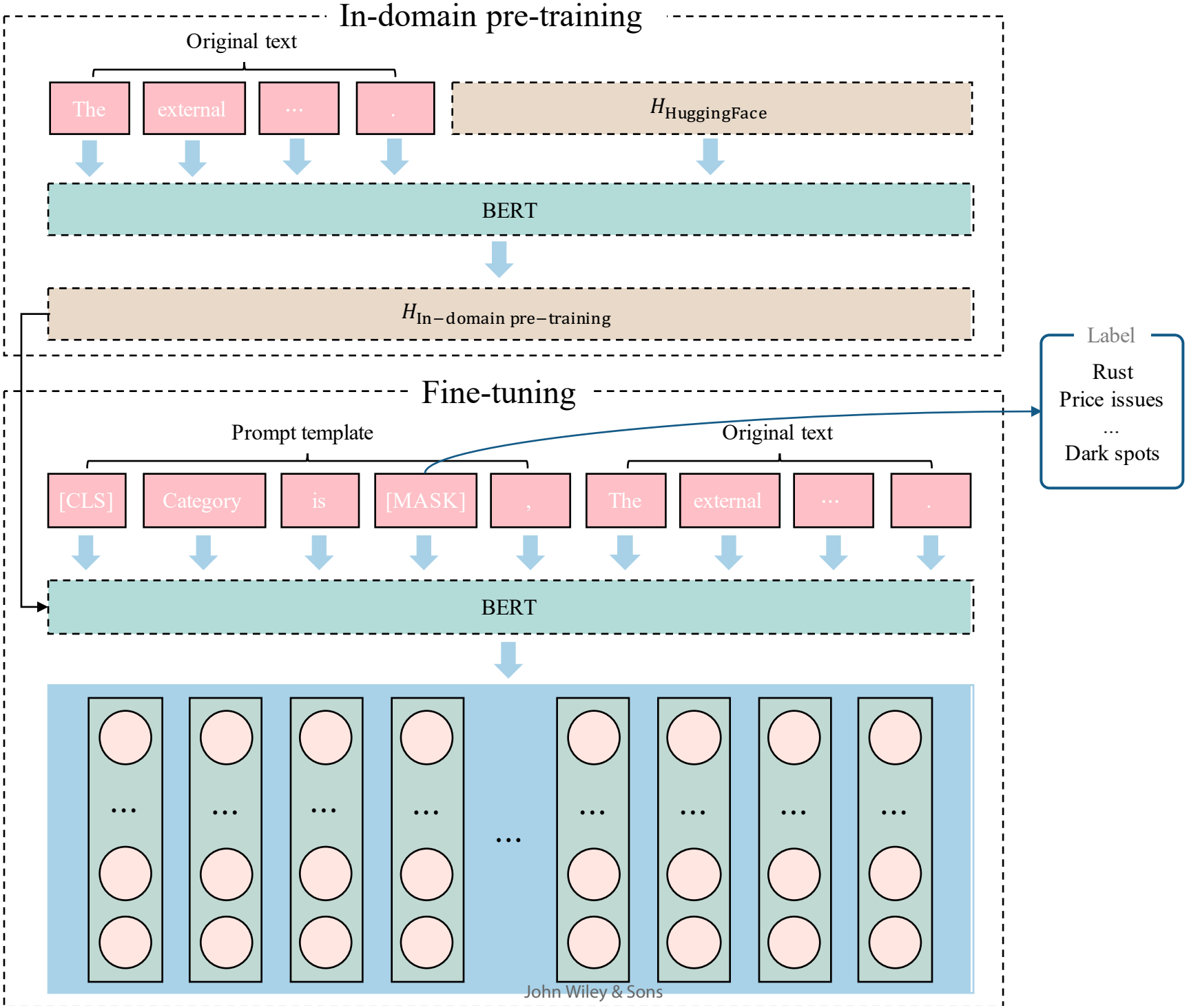
[19] Prechelt, L.: Early stopping - but when? In: Neural Networks: Tricks of the Trade - Second Edition vol. 7700, pp. 53–67. Springer, New York, NY, USA (2012)

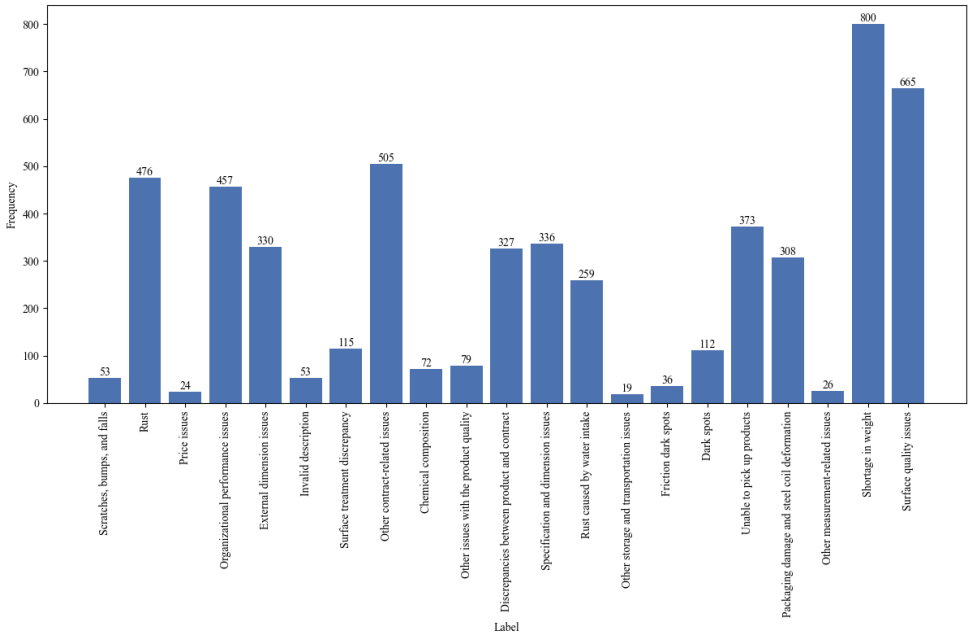
[20] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)

[21] Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., Xu, B.: Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, vol. 2, pp. 207–212. The Association for Computer Linguistics, Berlin, Germany (2016)

[22] Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, pp. 427–431. Association for Computational Linguistics, Valencia, Spain (2017)

[23] Johnson, R., Zhang, T.: Deep pyramid convolutional neural networks for text categorization. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 562–570. Association for Computational Linguistics, Vancouver, Canada (2017)





801x521mm (39 x 39 DPI)