Kalpa Publications in Computing

Volume 22, 2025, Pages 543–557

Proceedings of The Sixth International Conference on Civil and Building Engineering Informatics

Automated Text Classification of Construction Inspection Report: A Small Samples Training Approach

Kai Li[1*] ,Chao Dong[2†], Xueqing Fang[3‡] and Da Li[4§]

[1] Master Student, Department of Civil Engineering and Architecture, Wuhan University of Technology, Wuhan, Hubei, China.
Email: 2664361126@qq.com

[2]Ph.D., Assoc. Prof., Department of Civil Engineering and Architecture, Wuhan University of Technology, Wuhan, Hubei, China.
Email: chaodong@whut.edu.cn

[3] Master Student, Department of Civil Engineering and Architecture, Wuhan University of Technology, Wuhan, Hubei, China.
Email: 2545711470@qq.com

[4] Ph.D., Assoc. Prof., School of Civil Engineering, Henan University of Science and Technology, Zhengzhou, Henan, China.
Email: 9905639@haust.edu.cn

Abstract

Risk management is crucial for construction safety, but safety risk assessment often relies on experts' knowledge, which makes automatic risk management in engineering projects still a big challenge. Fortunately, for large-scale infrastructure construction, on-site inspection is required, and the conditions on-site are recorded in text format, which provides an opportunity to learn risk information from inspection reports. To improve document processing efficiency, automatic text classification plays an important role. However, currently, automatic text classification requires large scale training datasets. It is a big challenge for the engineering industry, especially for the fields which heavily rely on the experts' knowledge, such as risk assessment. Limited data sources, high time and labor costs make it not practical to establish a large-scale dataset. This work proposes a BERT-based ensemble model for small-sample text classification, leveraging the Focal loss function to address data imbalance issues. Concurrently, an ensemble strategy is employed to enhance the model's generalization capabilities, while the learning rate

[*] Kai Li created the first stable version of this document
[†] Chao Dong created the first draft of this document
[‡] Xueqing Fang created the first stable version of this document
[§] Da Li completed some auxiliary work

gradient descent method is applied to mitigate the risk of model overfitting. The efficacy of the proposed framework is validated through a four-classification task about identifying risk levels based on the inspection reports of a metro construction project. The BERT-based ensemble model proposed in this paper achieves an accuracy of 96.24% on the test set, surpassing other pre-trained classification models and excelling in automated text classification tasks.

1 Introduction

Construction is an inherently complex process characterized by long construction cycle, unique on-site conditions, sophisticated construction techniques, and significant environmental dependency. Compared to other industries, accidents happen frequently in the construction sector, leading to casualties and property damage. In 2017, the U.S. Bureau of Labor Statistics reported over 950 fatal injuries and more than 200,000 non-fatal injuries in the construction industry, accounting for more than 21% of all occupational fatalities in the United States (Bureau of Labor Statistics & Occupational Safety and Health Administration , 2018). To mitigate the harm caused by accidents, implementing safety risk management during construction process is crucial.

The primary objective of safety risk management is to assess the safety risk status of construction projects, which helps allocating appropriate resources to mitigate risk. Currently, safety risk assessment in construction industry still heavily relies on experts' experience. For example, for the risk assessment during foundation pits excavation, safety management engineers conduct on-site exploration daily, inspecting the on-site conditions (support structures, surrounding environment, and monitoring facilities etc.). Then, based on the exploration results and monitoring data, a construction site inspection report is created. Finally, experienced engineers make a comprehensive assessment of the construction safety risk based on the inspection report. This process is time-consuming and heavily relies on the engineers' expertise. The construction industry is labor-intensive, the shortage of experienced engineers, and the lengthy transmission time of risk information further hampers timely response to risks. Therefore, an automated method for safety risk assessment based on inspection reports needs to be developed to enable timely risk response.

The rapid advancement of artificial intelligence provides new possibilities for construction safety risk management, with Natural Language Processing (NLP) technologies demonstrating high accuracy and applicability in processing textual data (Kim & Chi, 2019). To address the limitations of manual risk assessment, automated methods for safety risk assessment in construction projects have begun to emerge. Automated safety risk assessment based on construction inspection is a text classification problem. Currently, the most popular automated text classification methods can be classified into Shallow Machine Learning models and Deep Learning models. Abderrahim Zermane adopted Random Forest model to classify the causes of falls from height (Zermane & Mohd Tohir, 2023). Fan Zhang et al. proposed an approach integrating multiple machine learning models to classify the causes of construction accident reports (Zhang et al., 2019). However, these algorithms have limited learning capabilities, require manually provided features, and exhibit high error rates. In contrast, deep learning algorithms can identify features automatically. Zhang et al. enhanced a CNN model by introducing multi-channel input to classify unstructured construction quality records (Zhang, Li, Tian, Song, & Shen, 2022). Similarly, Baker et al. employed a CNN and a hierarchical attention network (HAN), incorporating RNNs, to automatically extract accident precursors from construction accident report datasets (Baker, Hallowell, & Tixier, 2020). Even many attempts have been made in text mining in the construction industry, several challenges remain: (1) Semi-structured or unstructured natural language reports make it difficult for deep learning models to achieve a high accuracy; (2) Limited data sources and the high cost of manual labeling makes it a big challenge to develop large-scale domain-specific

datasets in construction industry; (3) Even construction domain is high risk, severe risk incidents are still rare, leading to data imbalance issues. Given these circumstances, the methods mentioned earlier struggle to handle text classification for imbalanced small-sample datasets in the construction domain.

In recent years, the emergence of BERT pre-trained models has offered a promising solution for addressing challenges associated with small datasets. As a pre-trained model, BERT requires minimal domain-specific data for fine-tuning to achieve the desired results. Thus, this paper proposes a BERT-based ensemble model. First, data augmentation techniques are employed to enhance sample diversity. Then, leveraging BERT's powerful text processing capabilities, features are extracted from construction inspection reports, followed by fine-tuning multiple models. The final prediction result is determined by aggregating the predictions from these models. Furthermore, to address the data imbalance issues inherent in small-sample datasets, the Focal Loss function is employed to mitigate the impact of class imbalance. Ultimately, an automated warning level identification mechanism for construction inspection reports is developed, assisting project managers conducting timely risk control.

2 Literature Review

2.1 Application of Text Classification in Construction Safety Management

Nowadays, there are many studies on automated text classification in the construction field. Typical text classification methods can be divided into knowledge-based methods and machine learning-based methods. Ontology is considered one of the most common knowledge-based approaches. Seokho Chi et al. developed an ontology-based text classification method to support automated job hazard analysis, identify key risk factors associated with each construction incident, and determine the critical risk combinations leading to accidents (Chi & Han, 2013). Salama and El-Gohary proposed a semantic machine learning-based text classification algorithm for categorizing clauses and sub-clauses in textual documents, facilitating compliance checking for construction regulations and contracts (Salama & El-Gohary, 2016).

With advancements in machine learning, new possibilities have emerged for text classification. For instance, Yang Miang Goh applied algorithms such as Support Vector Machine (SVM), Logistic Regression (LR), Random Forest (RF), K-Nearest Neighbors (KNN), Decision Tree (DT), and Naive Bayes (NB) to classify construction incident narratives. The F1 score of SVM ranged from 0.45 to 0.92, outperforming other classifiers (Goh & Ubeynarayana, 2017). Building on this, Fan Zhang et al. proposed an ensemble model based on multiple machine learning algorithms to classify the causes of construction accident reports (Zhang, Fleyeh, Wang, & Lu, 2019).

However, a limitation of machine learning is the requirement for manual feature extraction. Compared to traditional machine learning, deep learning is an end-to-end process capable of automatically learning features from training datasets. Today, research on text classification using deep learning is increasingly prevalent. In the construction domain, Botao Zhong et al. proposed a method combining Natural Language Processing (NLP) and Convolutional Neural Networks (CNN) to analyze and classify construction incident texts, while using an LDA model to explore the intrinsic relationships between different categories of accident causes (Zhong, Pan, Love, & Ding, 2020). Dan Tian et al. employed a CNN-based text classification model to categorize textual descriptions of construction site conditions into six classes, demonstrating the proposed model's reliability and applicability in handling large-scale construction site texts (Tian, Li, Shi, Shen, & Han, 2021). Hrishikesh Gadekar and Nikhil Bugalia proposed a semi-supervised YAKE-GLDA method for the automatic classification of construction safety reports. However, the YAKE-GLDA method is suitable only for medium-sized databases and not for smaller ones (Gadekar & Bugalia, 2023). Although there have been studies on

text classification in the construction field, there is still a gap in research on small-sample text classification.

2.2 Application of BERT in Text Classification

In most cases, obtaining large domain-specific datasets is challenging. Google has released a new language representation model based on the Transformer, Bidirectional Encoder Representations from Transformers (BERT) (Devlin, Chang, Lee, & Toutanova, 2018). BERT is pre-trained on a large corpus and fine-tuned by adding additional output layers. Therefore, advanced models built on BERT can accomplish NLP tasks without the need for training on large-scale datasets.

BERT has been successfully applied across various domains, demonstrating outstanding performance in text classification tasks. Chi Sun investigated a series of fine-tuning approaches, providing a general framework for fine-tuning BERT across different text classification tasks (Sun, Qiu, Huang, & Xu, 2019). Fang et al. developed a novel BERT-based model for the automatic classification of near-miss information in safety reports. The model was validated using a database of near-miss incident reports from real projects, achieving a 10% increase in accuracy after fine-tuning (Fang, et al., 2020).

Several studies have enhanced model performance by modifying BERT's classification head and integrating it with other neural networks. Kamaljit Kaur et al. proposed a Bidirectional Encoder-Decoder Transformer-Convolutional Neural Network (BERT-CNN) model for requirement classification, which improves model performance by stacking convolutional layers on top of the BERT layer. Experiments conducted on the PROMISE dataset containing 625 requirements demonstrated that the proposed model outperformed state-of-the-art baseline methods (Kaur & Kaur, 2023). Nishant Rai attempted to connect the output layers of an LSTM model and a BERT model for fake news classification of news headlines. Testing on the PolitiFact and GossipCop datasets showed an improvement in accuracy (Rai, Kumar, Kaushik, Raj, & Ali, 2022).

When facing challenging tasks or insufficient training data, ensemble methods are often employed to enhance model performance. J. Briskilal integrated BERT and RoBERTa models for the classification of idiomatic and literal text (Briskilal & Subalalitha, 2022). Rohan S (Baker, Hallowell, & Tixier, 2020) ingh Wilkho utilized the Bagging ensemble method, combining different models to achieve superior predictive performSance. He fine-tuned 50 models for each architecture type on different subsets of the training-validation set and experimentally determined the optimal number of models to ensemble (Wilkho, Chang, & Gharaibeh, 2024).

Based on previous works, we propose a BERT-based ensemble model for classifying imbalanced small-sample text datasets. This model can automatically assess the risk status of construction sites based on monitoring reports.

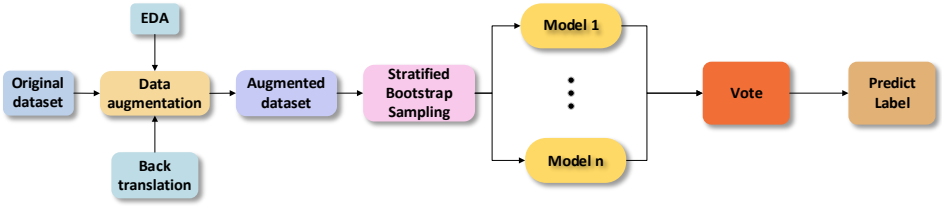

Figure 1: BERT-Based Ensemble Model Framework for Small-Sample Text Classification

3 Model Architecture

Our model is composed of multiple BERT classification models. Each model consists of a pre-trained BERT model and a classification head, both of which are fine-tuned on our dataset. The specific architecture is shown in Figure 1. The model takes a piece of text as input and outputs a label based on the content of the text. The following section introduces our model architecture.

3.1 Data Augmentation

This research focuses on text classification problems for small-sample datasets. To increase sample diversity, we used a data augmentation strategy combining back-translation and Easy Data Augmentation (EDA). EDA is a method for expanding text data through simple operations such as synonym replacement, random insertion, random swapping, and random deletion. By making slight modifications to the original text, the generated samples help the model learn richer textual expressions, improving its robustness. Compared to other data augmentation methods, this approach is simple, requiring no complex algorithms or extensive computational resources, while increasing sample diversity at minimal cost.

3.2 Bagging Ensemble

We used the Bagging ensemble method in our model. Bagging (Bootstrap Aggregating) is a widely used ensemble learning method aimed at improving machine learning models by combining the predictions of multiple models, reducing the risk of overfitting, and enhancing generalization. We trained different classifiers on various subsets of the same training dataset, with these subsets generated from the original dataset using Bootstrap sampling (which allows samples to be selected multiple times). After creating the subsets, we trained multiple models on them, which together formed the Bagging ensemble model. The final prediction is made through a simple majority voting process.

3.3 BERT Classification Model

Our model is built on the integration of multiple BERT classification models. This section focuses on explaining the individual BERT models in the ensemble, with the architecture of a single model shown in Figure 2. In each individual model, we use the Focal loss function to address the issue of data imbalance, and apply a learning rate decay strategy along with the AdamW optimizer to enhance model accuracy. Further explanation of the individual models will be provided in later sections.

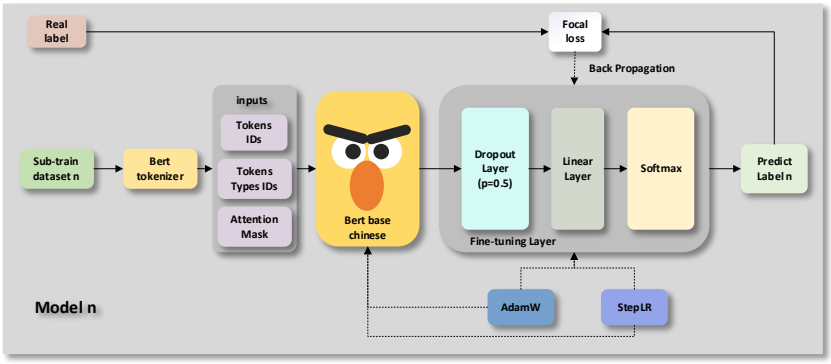

Figure 2: Structure of an individual BERT classification model

(1) Input Embedding

BERT's input embedding consists of three main components: token embedding, segment embedding, and position embedding. These embeddings work together to transform the text into a numerical form that the model can process. The BERT model only accepts fixed-length input sequences (with a maximum length of 512 tokens). We have fixed the input sequence length to 512 tokens, padding shorter sequences with empty tokens.

(2) Multi-layer Bidirectional Transformer Encoder

The multi-layer bidirectional Transformer encoder in BERT is the core of the model, responsible for processing text data through the input embedding layer. The encoder is based on the Transformer architecture and is composed of multiple identical stacked layers. Each layer consists of two main components: a multi-head self-attention mechanism and a position-wise feed-forward network.

(3) Classification Head

We use a pre-trained BERT model as a feature extractor, and add a fully connected layer with a softmax activation function on top for classification. The fully connected layer and activation function together form the classification head of the model, responsible for converting BERT's high-dimensional feature representation into the final classification prediction. Through the fully connected layer, the features extracted by BERT are mapped to the final classification labels. The output dimension of this layer equals the number of categories in the classification task. To improve the model's generalization ability, dropout is applied before the fully connected layer, randomly "dropping" some neuron outputs to reduce overfitting. Finally, the softmax function is used to calculate the predicted probability for each category. Additionally, Focal Loss is used to calculate the loss during model training, optimizing the process

4 Experiments

4.1 Dataset

Our dataset consists of 158 monitoring reports from a metro construction project, with experts manually labeling four warning levels: no warning, yellow warning, orange warning, and red warning. To increase sample diversity, we used back-translation and EDA methods for data augmentation. Table 1 shows the sample count for each warning level. The data is highly imbalanced, so we have applied two methods to cope with data imbalance: the Focal Loss function and the Bagging ensemble. When using BERT for text classification, preprocessing steps such as tokenization, adding special tokens, generating Token IDs and Type IDs, and constructing attention masks are necessary to meet the model's input requirements.

Dataset / Labels	Original Dataset	Augmented Dataset
No Warning	24	48
Yellow Warning	119	238
Orange Warning	9	108
Red Warning	4	48

Table 1: The distribution of labels in the dataset

4.2 Training, Validation, Testing

This paper applies fully supervised training of the BERT model on a small dataset, fine-tuning task-specific patterns to improve training speed and accuracy. By analyzing weights and the attention mechanism, the model provides interpretability for small datasets. The data is split into two groups: 70% for training and 30% for testing, with the training set further divided into two groups: 80% for training and 20% for validation.

We used the training set to perform bagging ensemble learning. We split the training dataset into 5 different subsets to train 5 different models (for each architecture).

The training and validation of BERT involve the fine-tuning process. The general fine-tuning parameters are: training epochs = 8, initial learning rate = 8e-6, optimizer = AdamW, and the loss function is FocalLoss with class weights of 5:1:30:1.

(1) Focal Loss Function

The cross-entropy loss function is commonly used when training deep learning models for classification, as it measures the difference between two probability distributions for a given random variable or set of events. However, this research focuses on imbalanced datasets. In such cases, training with the cross-entropy loss function can lead to models that disproportionately favor the majority class, resulting in suboptimal performance for the minority class. To mitigate this issue, the focal loss function is employed in this paper as a replacement for the cross-entropy loss during model training. Focal Loss function is calculated using Equation (1).

$$FL(p_t) = -\alpha_t(1-p_t)^{\gamma} log(p_t) \quad (1)$$

Where P_t corresponds to the predicted probability of the true class, α_t denotes the modulating factor to handle class imbalance. γ is a parameter that adjusts the attention given to easy and hard samples. Overall, the focal loss function addresses data imbalance by modulating the contribution of each sample to the loss. α_t alleviates the imbalance from the perspective of sample class by adjusting the weight of the loss calculation for different classes, while γ enhances the contribution of hard-to-learn samples to improve model performance. Typically, $\gamma = 2$ is set to a standard value in the field of computer vision.

(2) Comparative experiment and sensitivity analysis

To validate the effectiveness of Focal Loss function, comparative experiments were conducted. The results show that, compared to the traditional cross-entropy loss function, Focal Loss function demonstrates stronger classification performance on imbalanced datasets (see Table 2). In addition, we also studied the impact of the α_t parameter on the model's classification performance. With $\gamma = 2$, the α_t parameter was systematically adjusted in the range from 0 to 30, with intervals of 10, and the classification performance was recorded after each training session. The experimental results show that as the α_t value increases, the model's classification performance on imbalanced classes improves, further confirming the model's sensitivity to the α_t parameter. This finding suggests that selecting an appropriate α_t parameter helps improve the model's ability to identify minority classes.

(3) Learning Rate Decay

Learning rate is a key hyperparameter that controls how the model's weights are adjusted. An appropriate learning rate can speed up convergence and avoid oscillations or overly slow progress. A traditional constant learning rate often fails to meet the needs of deep networks, requiring numerous experiments and making it difficult to find the optimal value. This paper uses a time decay strategy, starting with a larger learning rate to quickly approach the optimal solution, and then gradually reducing it to allow fine-tuned adjustments in later stages and avoid oscillations.

For small datasets, we used cross-validation to determine the hyperparameters. After multiple tests, the learning rate was reduced to 90% of its original value after each epoch, achieving the highest accuracy, as shown in Figure 3.

Types of Loss Functions	Index	No warning	Yellow warning	Orange warning	Red warning
cross-entropy loss function	F1score	0.8959	0.9270	0.8616	1
Focal Loss function (class weight =1:1:1:1)	F1score	0.8959	0.9329	0.8808	1
Focal Loss function (class weight =1:1:10:1)	F1score	0.8887	0.9404	0.8925	1
Focal Loss function (class weight =1:1:20:1)	F1score	0.9280	0.9464	0.8925	1
Focal Loss function (class weight =1:1:30:1)	F1score	0.9280	0.9592	0.9356	1

Table 2: Comparative experiment and sensitivity analysis of Focal Loss function

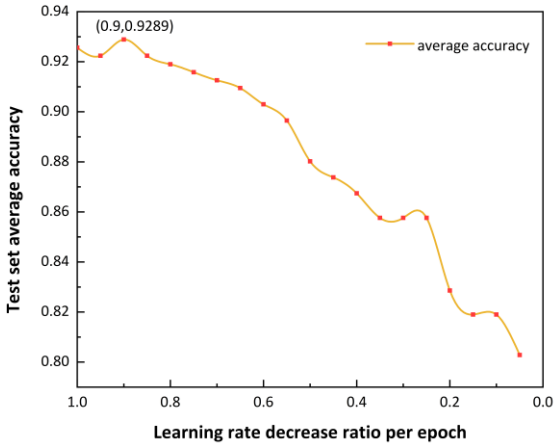

Figure 3: Optimal learning rate decay ratio

4.3 Evaluation Metrics

Since our research involves a multi-label classification problem, it is important to evaluate not only the overall performance of the model, but also its ability to correctly identify each label. Therefore, we use the F1 score to assess the model's performance on each individual label and the micro-F1 score to evaluate the model's overall performance across all labels. The F1 score is calculated using Equations (2), (3), and (4).

$$\text{Precision}=\frac{TP}{TP+FP} \tag{2}$$

$$\text{Recall}=\frac{TP}{TP+FN} \tag{3}$$

$$F1\text{score}=2\times\frac{\text{precision}*\text{Recall}}{\text{precision}+\text{Recall}} \tag{4}$$

5 Case Study

To validate the effectiveness of the proposed small-sample text classification framework in automatic text classification, we created two datasets. The data samples are shown in **Table 3**. The model's performance was evaluated through 7-fold cross-validation, and we compared our model with other models suitable for small-sample classification as benchmarks.

ID	Label	Criteria	Narrative example
0	no warning	Based on the four aspects included in the inspection daily reports—construction conditions, support structure, surrounding environment, and monitoring facilities—experts were asked to score the reports and classify the foundation pit risk warning levels into four categories.	Narrative: 开挖面土体为中密卵石，基坑开挖区域为 0-24 轴，1-8 轴制作顶板模板，8-12 轴中板浇筑，12-16 轴搭设中板脚手架，16-19 轴制作底板垫层，19-24 轴已开挖至设计深度，停止开挖，第四层钢支撑、第三层钢支撑 1-18 轴、第二层钢支撑 1-10 轴已拆除，东端盾构井有渗水，系用坑内降水，降水设施运转正常，支护状体无裂缝、无明显缺陷、隆起，基坑侧壁及坑内无涌水、流沙、管涌，基坑周围无超载。支护桩无裂缝、侵陷等情况，冠梁已施工完成，连续性完好，无过大变形、裂缝基坑防坠措施完好，钢支撑架设及时。基坑周边土体存在细小裂缝、车辆碾压等情况 ，无明显沉陷隆起。地表竖向位移监测点 DBC29-01、DBC29-02 等破坏，桩顶位移监测点 ZQS27 无法观测。 Label: yellow warning
1	yellow warning		
2	orange warning		
3	red warning		

Table 3: Labels used for the data, their criteria and sample narrative

5.1 Summary of the data set

For training and testing purposes, we divided 156 original monitoring inspection reports from a metro project into a total training dataset and a test dataset in a 7:3 ratio. The total training dataset was further split into a training set and a validation set at a ratio of 8:2. The inspection reports are formulated by manual daily inspection about construction conditions, support structure, surrounding environment, and monitoring facilities, which provide a comprehensive view of the foundation pit's risk conditions.

We invited experts with safety management experience over 15 years in metro construction, to label inspection reports by evaluating risk levels based on each inspection report. The risk levels are divided into four categories: no warning, yellow warning, orange warning, and red warning. No warning indicates that the project is in a low-risk state, and risk response measures are generally not required. Yellow warning indicates that the project's risk level is slightly higher than no warning, requiring minimal resources to control the risk. Orange warning indicates that the project is in a medium-high risk state, necessitating considerable resources and proper risk control measures to control the risk. Red warning indicates that the project is in a high-risk state, which typically leads to severe engineering accidents and requires close attention from project managers. Due to the specific nature of the construction industry, serious risks are undesirable. Reasonable risk control measures are often taken before risk evolves to high levels, leading to significant data imbalance. Normal and yellow warning samples make up the majority, while orange and red warnings are rare. Additionally, our dataset faces the issue of being a small sample dataset.

To improve the model's training performance and reliability, we applied data augmentation method to original dataset. In the following sections, we will use both the original data and the augmented data to train and evaluate the model separately.

5.2 Model Performance Based on Small-Sample Training

The BERT ensemble model was implemented using the Pytorch deep neural network platform. The hyperparameters of the deep neural network were determined based on the best results from multiple rounds of experimental testing. The specific parameters are shown in Table 4.

80% of the training set was used for model training, while the remaining 20% was used for model validation. The training loss from the original dataset is shown in Figure 4. After 20 epochs, the training loss fluctuated slightly, indicating that the model had reached optimal performance on the current dataset. In the final stage of training, the recall rate for the predefined labels stabilized at a relatively steady level. After 25 epochs of training, the recall rates for the normal and yellow warning labels stabilized above 0.9. Although there were only 13 samples for the normal label, the prediction accuracy was high, indicating that reports without warnings were more clearly expressed. However, due to significant data imbalance in the original training dataset, the red and orange warning samples were scarce (2 red samples and 4 orange samples), making them hard-to-learn samples and resulting in considerable fluctuations during training.

Due to the limited sample size, we conducted a 7-fold stratified cross-validation to verify the reliability of the trained model. We randomly divided the total training set consisting of 109 samples into 7 parts, aiming to maintain the original sample distribution as much as possible. In each round, one part was selected as test set, while the remaining 6 parts were combined as the training data set. Figure 5 (a, b) shows the precision and recall distributions of the classification labels from the 7-fold cross-validation. Since the number of orange and red warning samples in the dataset is extremely limited, and the distinction between orange and yellow warnings is not clear, random partitioning cannot ensure that each fold in cross-validation contains all four label categories. Therefore, in Figure 5 (a, b), the lowest training precision and recall rates for the orange and red warning labels can be zero.

The precision and recall distributions from the 7-fold cross-validation show that the range for orange and red warning labels is quite large, while the precision and recall for no warning and yellow warning labels are distributed at a higher level. The training results indicate that the size of the training dataset directly affects the model’s robustness. Due to the lack of training samples, the model's robustness is weak, especially for sample categories with lower proportions in the training dataset. However, the metrics for no warning and yellow warning labels are distributed at a high level, indicating that the model can perform text classification tasks well once the dataset reaches a sufficient size.

5.3 Model Performance Based on Data-Augmented Training

In the initial stage, the total training dataset contained only 109 samples, with very limited samples for each category. Therefore, we expanded our dataset using a combination of EDA and back-translation data augmentation methods. The expanded training dataset includes 309 samples.

Training on the augmented dataset improved the model's robustness. As shown in Figure 4, after 15 training epochs, the training loss stabilized without fluctuation. The recall rate during the training process stabilized after 10 epochs. The expanded training dataset made the model training more stable, with faster convergence, and there were no significant fluctuations in any class labels during training. Evaluating the model's performance in the final training stage, the recall rates for all class labels exceeded 0.9, indicating a significant improvement in the model's training performance.

To verify the model's robustness, a 7-fold cross-validation was performed on the augmented dataset, with precision and recall rates shown in Figure 5 (d, e). Except for the orange warning, the average

precision and recall rates for the other warning categories were above 0.95. Although the recall rate for the orange warning fluctuated, its average reached 0.8.

Hyperparameters	Value
Batch Size	2
Dropout Ratio	0.5
Optimizer	AdamW
Initial learning rate	8e-6
Optimal number of integration models	5

Table 4: Bert integrated model training parameters

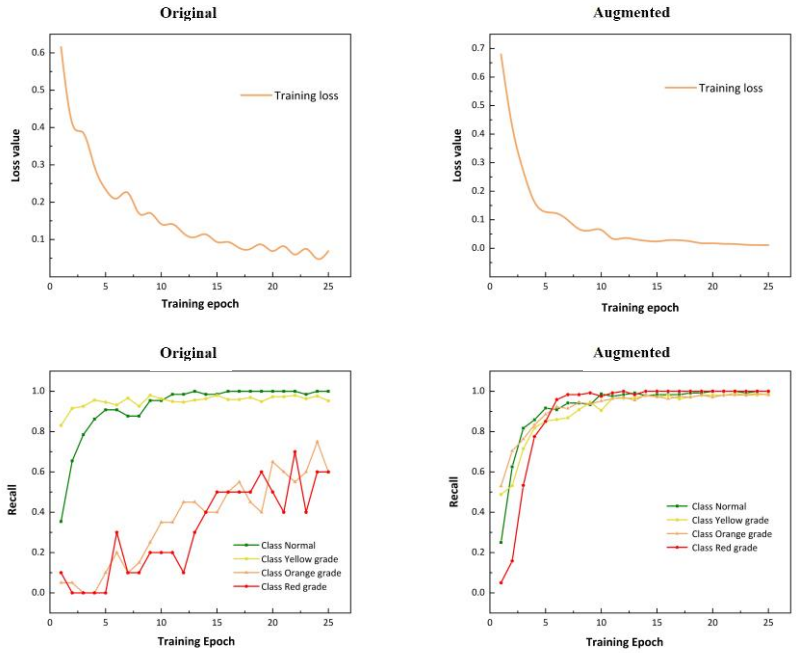

Figure 4: Training loss curve and Training recall curve

5.4 Model's Robustness Validation

To accurately evaluate the model's performance, we validated the model on a completely new test set, and the test results are shown in Figure 5 (c, f). On the original dataset, although the overall test accuracy reached 87.23%, for the orange warning category, all 3 test samples were predicted incorrectly, resulting in an accuracy of 0 for this label. As for the red warning samples, although they were all predicted correctly, there was only one red sample in the original test set, so the accuracy result lacks statistical significance due to the small sample size.

On the augmented dataset, we clearly saw that although the overall accuracy maintains 96.24%, the model also achieved high accuracy for each individual category, with only few yellow and orange samples incorrectly predicted. While ensuring high accuracy, the augmented test set contains enough samples for each category, making the results more convincing.

To examine the advantage of the model in small-sample text classification, we compared the ensemble model with other BERT-based text classification models. We trained and tested the models using both the original dataset and the augmented dataset. For a more comprehensive evaluation, we calculated the precision, recall, and F1 score for each category in the test set, and the results are shown in Table 5.

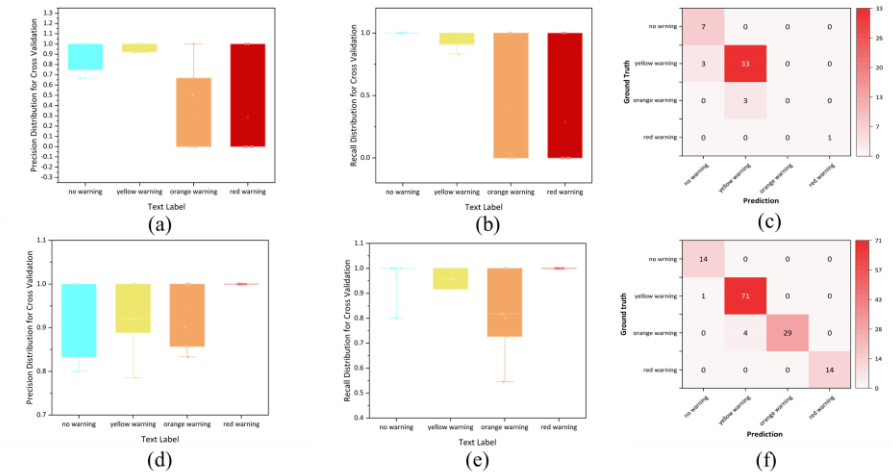

Figure 5: Performance comparison of the model on the original (a, b, c) and augmented (d, e, f) datasets

First, the models were trained on the original dataset. For the normal and yellow warning categories, the F1 scores of the models were almost identical and performed well. Due to the small number of orange samples, all models performed poorly in this category, but BERT + DPNN and BERT + RCNN showed some improvement in predicting the orange warnings. Although the number of red warning samples was also small, all models performed well in identifying the red warnings.

On the augmented dataset, the F1 scores of all models are relatively high, indicating that each model performs well in classifying imbalanced small samples. However, the F1 scores of the ensemble model are higher than those of the other models, demonstrating that our proposed ensemble model offers significant advantages and better performance in small-sample text classification tasks.

6 Discussion

Small-sample datasets are prone to overfitting during model training, making it difficult to learn features accurately. This is often accompanied by data imbalance issues, as seen in our dataset, where the model tends to favor the majority classes and ignore the minority classes. As shown in Figure 4, during the convergence process, the model's training loss on the original dataset did not stabilize at a low level in the later stages and exhibited some fluctuations. The recall for the orange and red warning categories fluctuated significantly during training. On the original dataset, as shown in Figure 5 (c), the ensemble model achieved an overall test accuracy of 87.23%, but for the orange warning category, all 3 test samples were predicted incorrectly, resulting in an accuracy of 0 for this category. As for the red warning samples, all were predicted correctly, there was only one sample, making the result less reliable due to randomness. To address the issue of insufficient data, we applied data augmentation techniques to expand our dataset, increasing sample diversity and providing more learnable samples. During the convergence process, compared to the original dataset, the model's training loss on the augmented

dataset was smoother and stabilized at a lower level, with recall for all categories converging to 0.95, as shown in Figure 4. The generalization ability of the ensemble model was also improved, with the average metrics for all categories in 7-fold cross-validation exceeding 0.8. Additionally, the BERT-based model more accurately identified category features on the augmented test set.

In Section 5.4, when comparing with other models, the performance differences between models on the original dataset were small, mainly due to the limited dataset size, which restricted the models' capabilities. However, on the augmented dataset, the ensemble model outperformed other models on all metrics, indicating that our multi-model ensemble architecture and the Focal Loss function performed well in coping with data imbalance in small datasets.

However, there are certain limitations to our research. First, the model training still needs manual labels, which is a time-consuming process. With the accumulation of domain-specific datasets, manual work can be reduced with the help of unsupervised training on large datasets. Additionally, we only trained and tested the model's effectiveness on the inspection reports from one real-world project. Future work will be requires testing the model on other text classification problems in the construction industry.

Model name	Index	No warning		Yellow warning		Orange warning		Red warning	
		Original	Augmented	Original	Augmented	Original	Augmented	Original	Augmented
BERT-based ensemble model	Precision	**0.7000**	**0.9333**	**0.9167**	**0.9467**	**0.0000**	**1.0000**	**1.0000**	**1.0000**
	Recall	**1.0000**	**1.0000**	**0.9167**	**0.9861**	**0.0000**	**0.8788**	**1.0000**	**1.0000**
	F1score	**0.8235**	**0.9655**	**0.9167**	**0.9660**	**—**	**0.9355**	**1.0000**	**1.0000**
BERT+ CNN	Precision	0.7000	0.8235	0.9167	0.8718	0.0000	0.9583	1.0000	1.0000
	Recall	1.0000	1.0000	0.9167	0.9444	0.0000	0.6970	1.0000	1.0000
	F1score	0.8235	0.9032	0.9167	0.9066	—	0.8070	1.0000	1.0000
BERT+ RNN	Precision	0.7000	0.8235	0.9167	0.9571	0.0000	0.9375	1.0000	1.0000
	Recall	1.0000	1.0000	0.9167	0.9306	0.0000	0.9091	1.0000	1.0000
	F1score	0.8235	0.9032	0.9167	0.9461	—	0.9231	1.0000	1.0000
BERT+D PNN	Precision	0.7000	0.9286	0.9697	0.9583	0.6667	0.9375	1.0000	1.0000
	Recall	1.0000	1.0000	0.8889	0.9583	0.6667	0.9091	1.0000	1.0000
	F1score	0.8235	0.9630	0.9275	0.9583	0.6667	0.9231	1.0000	1.0000
BERT+R CNN	Precision	0.7000	0.8235	0.9412	0.9189	1.0000	0.9643	1.0000	1.0000
	Recall	1.0000	1.0000	0.9143	0.9444	0.3333	0.8182	1.0000	1.0000
	F1score	0.8235	0.9032	0.9276	0.9315	0.5000	0.8853	1.0000	1.0000

Table 5: Comparison of training performance across different models

7 Conclusion

During construction, a large amount of unstructured and semi-structured text is generated, providing project managers with valuable information. However, reading and identifying useful information in inspection reports and classifying warning levels is typically a manual and time-consuming process. Additionally, due to the limited data sources and the high cost of manual labeling, it is difficult to form large-scale domain datasets.

In summary, this paper proposes a BERT ensemble model to address the issue of imbalanced small-sample datasets. First, a combination of EDA and back-translation data augmentation techniques is applied to enhance the diversity of the small-sample dataset without collecting additional data. The training dataset is then divided into multiple subsets, with each model trained on a different subset. The final prediction label is determined by a voting mechanism. For individual models, learning rate gradient descent is used to reduce the risk of overfitting, and the Focal Loss function assigns weights to different classes during loss calculation, improving the model’s ability to learn from minority, hard-to-learn samples. Finally, to verify the feasibility of the proposed model, the paper classifies real textual

data from inspection reports of a real project. The data was split into training and test sets, with cross-validation performed on the training set, showing that the model performs well on different datasets and has strong generalization capabilities. Additionally, the ensemble model achieved a classification accuracy of 96.24% on the test set, slightly outperforming other classification models and demonstrating good robustness.

This model framework provides a viable path for automated risk assessment from construction site inspection reports, supporting intelligent construction management, enabling timely responses to safety risks.

References

Baker, H., Hallowell, M. R., & Tixier, A. J.-P. (2020). Automatically learning construction injury precursors from text. *Automation in Construction, 118*, 103145*. doi:https://doi.org/10.1016/j.autcon.2020.103145

Briskilal, J., & Subalalitha, C. N. (2022). An ensemble model for classifying idioms and literal texts using BERT and RoBERTa. *Information Processing and Management, 59, 102756*. doi: https://doi.org/10.1016/j.ipm.2021.102756.

Bureau of Labor Statistics, & Occupational Safety and Health Administration . (2018). *Commonly used statistics.* doi:OSHA.gov. https://www.osha.gov/oshstats/commonstats.html

Chi, S., & Han, S. (2013). Analyses of systems theory for construction accident prevention with specific reference to OSHA accident reports. *International Journal of Project Management, 31(7), 1027-1041*. doi:https://doi.org/10.1016/j.ijproman.2012.12.004.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv Preprint*. doi:arXiv:1810.04805.

Fang, W., Luo, H., Xu, S., Love, P. E., D, L. Z., & Ye, C. (2020). Automated text classification of near-misses from safety reports: An improved deep learning approach. *Advanced Engineering Informatics, 44, 101060*. doi: https://doi.org/10.1016/j.aei.2020.101060.

Gadekar, H., & Bugalia, N. (2023). Automatic classification of construction safety reports using semi-supervised YAKE-guided LDA approach. *Advanced Engineering Informatics, 56, 101929*. doi:https://doi.org/10.1016/j.aei.2023.101929

Goh, Y. M., & Ubeynarayana, C. U. (2017). Construction accident narrative classification: An evaluation of text mining techniques. *Accident Analysis and Prevention, 108, 122-130*. doi: https://doi.org/10.1016/j.aap.2017.08.026.

Kaur, K., & Kaur, P. (2023). BERT-CNN: Improving BERT for requirements classification using CNN. *Procedia Computer Science, 218, 2604-2611*. doi: https://doi.org/10.1016/j.procs.2023.01.234

Kim, T., & Chi, S. (2019). Accident case retrieval and analyses: Using natural language processing in the construction industry. *Journal of Construction Engineering and Management, 145(3), 04019004*. doi:https://doi.org/10.1061/(ASCE)CO.1943-7862.0001625.

Rai, N., Kumar, D., Kaushik, N., Raj, C., & Ali, A. (2022). Fake news classification using transformer-based enhanced LSTM and BERT. *International Journal of Cognitive Computing in Engineering, 3, 98–105*. doi:https://doi.org/10.1016/j.ijcce.2022.03.003

Salama, D. M., & El-Gohary, N. M. (2016). Semantic text classification for supporting automated compliance checking in construction. *ournal of Computing in Civil Engineering, 30(1), 04014106*. doi:https://doi.org/10.1061/(ASCE)CP.1943-5487.0000301.

Sun, C., Qiu, X., Huang, X., & Xu, Y. (2019). How to Fine-Tune BERT for Text Classification? *Springer, Cham*. doi:arxiv.org/abs/1905.05583

Tian, D., Li, M., Shi, J., Shen, Y., & Han, S. (2021). On-site text classification and knowledge mining for large-scale projects construction by integrated intelligent approach. *Advanced Engineering Informatics, 49, 101355, 49, 101355*. doi: https://doi.org/10.1016/j.aei.2021.101355.

Wilkho, R. S., Chang, S., & Gharaibeh, N. G. (2024). FF-BERT: A BERT-based ensemble for automated classification of web-based text on flash flood events. *Advanced Engineering Informatics, 59, 102293*. doi:https://doi.org/10.1016/j.aei.2023.102293

Zermane, A., & Mohd Tohir, M. Z. (2023). Predicting fatal fall from heights accidents using random forest classification machine learning model. *Safety Science, 159, 106023*. doi:https://doi.org/10.1016/j.ssci.2022.106023

Zhang, D., Li, M., Tian, D., Song, L., & Shen, Y. (2022). Intelligent text recognition based on multi-feature channels network for construction quality control. *Advanced Engineering Informatics, 53, 101669*. doi:https://doi.org/10.1016/j.aei.2022.101669.

Zhang, F., Fleyeh, H., Wang, X., & Lu, M. (2019). Construction site accident analysis using text mining and natural language processing techniques. *Automation in Construction, 99, 238-248, 99, 238-248*. doi:https://doi.org/10.1016/j.autcon.2018.12.016

Zhong, B., Pan, X., Love, P. D., & Ding, L. (2020). Deep learning and network analysis: Classifying and visualizing accident narratives in construction. *Automation in Construction, 113, 103089*. doi:https://doi.org/10.1016/j.autcon.2020.103089