DeepBlueAI at PragTag-2023:Ensemble-based Text Classification Approaches under Limited Data Resources

Zhipeng Luo   **Jiahui Wang**   **Yihao Guo**
DeepBlue Artificial Intelligence Technology (Shanghai) Co., Ltd, Shanghai, China
{luozp, wangjh, guoyh}@deepblueai.com

Abstract

Due to the scarcity of review data and the high annotation cost, in this paper, we primarily delve into the fine-tuning of pretrained models using limited data. To enhance the robustness of the model, we employ adversarial training techniques. By introducing subtle perturbations, we compel the model to better cope with adversarial attacks, thereby increasing the stability of the model in input data. We utilize pooling techniques to aid the model in extracting critical information, reducing computational complexity, and improving the model's generalization capability. Experimental results demonstrate the effectiveness of our proposed approach on a review paper dataset with limited data volume.

1 Introduction

Peer review stands as a fundamental pillar of the scientific process, yet it presents formidable challenges that could greatly benefit from automation and support. At the heart of peer review are review reports – concise, argumentative documents in which reviewers assess research papers and offer recommendations for improvement. Automating the analysis of argumentation within peer reviews (Dycke et al., 2023) holds vast potential, ranging from facilitating meta-scientific investigations into review practices to consolidating insights from multiple reviews and aiding less experienced reviewers.

Text classification is a significant and challenging task. However, when relying on relatively small datasets, traditional machine learning methods may encounter issues such as overfitting and poor generalization performance. In such cases, pre-trained models serve as powerful tools that offer robust solutions for addressing data scarcity. Pre-trained models, particularly those based on the deep learning Transformer (Vaswani et al., 2017) architecture, have demonstrated significant success in natural language processing tasks.

RoBERTa (A Robustly Optimized BERT Pretraining Approach) (Liu et al., 2019), XLM-RoBERTa (Conneau et al., 2019) and DeBERTa (Deep BERT with Disentangled Attention) (He et al., 2023) are pre-trained models based on the Transformer architecture that have garnered widespread attention in the field of NLP. Through fine-tuning these pre-trained models, exceptional performance can be achieved on smaller datasets, mitigating overfitting issues and improving generalization performance.

This paper focuses on the application of RoBERTa, XLM-RoBERTa and DeBERTa to address text classification problems within a peer review dataset.

2 Related work

Pragmatic tagging of peer reviews is, in fact, a classification task, and in common classification tasks. In the field of text classification, models like Recurrent Neural Networks (RNN) (Jordan, 1997), and Long Short-Term Memory networks (LSTM) (Hochreiter and Schmidhuber, 1997) introduced more nonlinear factors, enabling them to automatically learn feature representations from data and achieving remarkable results.

However, deep learning methods may face overfitting issues on small datasets and require a substantial amount of labeled data for training. To address these issues, the development of pre-trained models has become a groundbreaking direction. Pre-trained models are trained on large-scale unlabeled corpora, learning rich language representations that enable them to better capture semantic relationships between words, as seen in models like BERT (Devlin et al., 2018). Subsequently, these pre-trained models can be fine-tuned for specific tasks to exhibit exceptional performance.

Among these models, RoBERTa (Liu et al., 2019) and DeBERTa (He et al., 2023) are representatives of pre-trained models based on the Trans-

Proceedings of the 10th Workshop on Argument Mining, pages 202–206
December 7, 2023. ©2023 Association for Computational Linguistics

	Recap	Strength	Weakness	Todo	Other	Structure
Count	87	62	130	245	106	109
Percentage(%)	11.77	8.39	17.59	33.15	14.34	14.75

Table 1: Number and percentage of each category in low_data

former architecture. RoBERTa achieved significant performance improvement across various NLP tasks by adjusting pre-training strategies and hyperparameters. On the other hand, DeBERTa enhanced the model's generalization capability and performance on different tasks by introducing disentangled attention mechanisms. In small dataset text classification tasks, models like RoBERTa and DeBERTa demonstrate remarkable capabilities. Their learned rich semantic representations from extensive corpora enable them to effectively extract features and capture relationships between sentences even in data-limited scenarios.

In the application of pre-trained models, researchers have introduced various techniques to further optimize model performance. Techniques such as k-fold cross-validation better evaluate the model's stability and generalization ability. Adversarial training methods like Fast Gradient Method (FGM)(Miyato et al., 2016) enhance the model's robustness, preventing it from being disrupted by adversarial attacks. Pooling techniques such as max pooling, min pooling and attention pooling allow models to understand text information at different levels. Additionally, model ensemble techniques combine predictions from multiple models, improving overall classification performance.

3 Task description

The goal of this task is to perform automatic analysis of argumentation in peer review. Our input data consists of each sentence in the argumentation, and the output results are the corresponding label categories for each sentence. The competition is divided into multiple stages, each providing two datasets(Kuznetsov et al., 2022)(Dycke et al., 2022): "low_data" and "full_data".

The training set provided in the "low_data" comprises a total of 34 review articles, 793 sentences in total. The objective is to classify each sentence into one of the six categories. We have conducted a statistical analysis for each category in the dataset, and the results are presented in Table 1.

The training set provided in the "full_data" consists of a total of 118 review articles, 2324 sentences in total. The objective remains the classification of each sentence into one of the six categories. Similar to the previous dataset, we conducted a statistical analysis for each category in the dataset, and the results are presented in Table 2.

4 Methodology

4.1 Model architecture

In this task, we primarily utilized three architecture-based pre-trained models: DeBERTa-v3-large, RoBERTa-large and XLM-RoBERTa-large, as our benchmark models. We incorporated a pooling layer to project features into lower dimensions, effectively reducing the number of parameters and computational load in the network while preserving essential information. Moreover, specific linear layers were added based on the number of task categories, yielding probabilities for each category. Ultimately, the highest predicted probability was selected to determine the final classification outcome of the model.

4.2 Pooling

In this section, we mainly used 2 types of pooling, attention pooling and maximum pooling, and ensembled the two different pooling models obtained when calculating the final result.

Attention Pooling: Attention pooling is a technique that enhances critical information while capturing local features in text. We calculate the weight for each token and effectively model relationships between different words. Specifically, the input word embedding sequence is weightedly aggregated and normalized, yielding a weight vector. This weight vector indicates the higher significance of specific words within the text. By element-wise multiplication of this weight vector with the word embedding sequence, we obtain the text representation after attention pooling.

Max Pooling: Max pooling is a common pooling technique employed to extract crucial features from local regions. In our approach, we apply max pooling to text representations to emphasize significant information within the text. Specifically, we perform max pooling operations on each window, se-

	Recap	Strength	Weakness	Todo	Other	Structure
Count	346	220	377	681	401	301
Percentage(%)	14.875	9.458	16.208	29.278	17.240	12.941

Table 2: Number and percentage of each category in full_data

lecting the maximum value within the window as the representation for that window. This technique aids in capturing key features in the text.

4.3 Adversarial Training

To enhance the model's robustness, we introduced adversarial training, specifically utilizing the Fast Gradient Method (FGM). FGM is an adversarial attack technique that we applied during the training process by injecting slight perturbations into the embedding layer. This compels the model to better handle adversarial attacks. Adversarial training in our approach involves computing the gradient of the loss function with respect to the input at each training iteration and slightly updating the input. By incorporating adversarial training, our approach elevates the model's robustness, enabling it to better handle interference within input data.

K-fold	bs = 2	bs = 4	bs = 4
P0(%)	83.538	82.384	80.533
P1(%)	80.711	82.431	82.29
P2(%)	78.298	83.859	91.789
P3(%)	88.55	90.106	73.363
P4(%)	85.141	81.899	79.638
P5(%)	-	-	87.8
P6(%)	-	-	84.755
P7(%)	-	-	79.021
P8(%)	-	-	82.161
Avg(%)	83.2476	84.1358	82.372

Table 3: Multifold cross-validation results for different models on low_data

4.4 K-Fold Cross Validation:

Model ensemble is a widely employed technique in machine learning competitions, while k-fold cross-validation serves as a common method to assess and enhance model performance during the training process. In k-fold cross-validation, the dataset is partitioned into k mutually exclusive subsets. Among these, k-1 subsets are utilized as training data, and the remaining subset serves as validation data. We iterate through k-fold cross-validation multiple times, each time selecting a different subset as the validation data. This ensures that each sample gets an opportunity to be used for validation. This way, we obtain k performance evaluation metrics, enabling a comprehensive understanding of the model's performance.

5 Experiments

5.1 Setting

On the "low_data" dataset, we fine-tuned various parameter values and selected the parameter combination that yielded the best experimental results. Specifically, the batch size was set to different values, namely 2 and 4, while the initial learning rate was set to 1×10^{-4}. Other configurations remained consistent with those used on the "full_data" dataset. For the "full_data" dataset, during the training process of all models, we set the batch size to 8 and the initial learning rate to 1×5^{-4}. Subsequently, a learning rate decay was applied, with a decay rate of 0.5 and a minimum of 1×10^{-7}. The models were trained for 10 epochs, with the early stopping strategy in place. Training would be stopped if the performance did not improve after 3 consecutive epochs. All training was conducted on V100-32G GPUs.

5.2 Training results on low_data

We recorded the results of k-fold cross-validation during the training process of the single DeBERTa-v3-large model on the "low_data" dataset. The batch size for the first experimental group was set to 2, while the subsequent two groups used a batch size of 4. For the first two groups of experiments, the dataset was divided into 5 subsets for training. In the third group, the dataset was split into 9 subsets for training. The interim results of training, as well as the average across folds, are presented in Table 3. Since the same model was employed, the first row of the table distinguishes solely based on the batch size used.

5.3 Training results on full_data

As depicted in Table 4, we have documented the k-fold cross-validation outcomes of model training on the "full_data" dataset. The models em-

K-fold	RoBERTa	RoBERTa(MaxPooling)	DeBERTa	DeBERTa	XLM-R	XLM-R(FGM)
P0(%)	86.116	86.452	86.846	87.037	86.920	84.518
P1(%)	83.246	85.259	86.379	85.024	80.433	86.149
P2(%)	87.273	90.455	90.991	91.699	90.519	89.422
P3(%)	81.627	84.676	84.919	82.878	81.76	81.183
P4(%)	83.13	84.020	85.513	83.994	81.852	83.624
P5(%)	81.005	85.078	87.435	87.106	84.208	81.121
P6(%)	85.932	85.915	88.821	86.818	85.833	86.397
P7(%)	83.861	85.112	84.617	82.779	84.290	85.753
P8(%)	82.959	82.026	84.044	82.713	82.863	80.679
P9(%)	81.775	83.326	82.336	-	82.722	84.827
Avg(%)	83.6924	85.232	86.190	85.561	84.14	84.3673

Table 4: Presentation of results at various stages

ployed in this study are RoBERTa-large, XLM-RoBERTa-large and DeBERT-v3-large. For the fine-tuning of RoBERTa-large, we adopted the max pooling approach, after applying the max pooling technique during fine-tuning, the avg_f1_mean score increased from 83.6924 to 85.2319. When fine-tuning with XLM-RoBERTa-large, we experimented with the inclusion of FGM. Compared to not using FGM, the avg_f1_mean score improved from 84.14 to 84.3673. When fine-tuning DeBERT-v3-large, we conducted two sets of experiments, both utilizing attention pooling techniques. The primary distinction between the first and second experiments lay in the use of 10-fold and 9-fold cross-validation, respectively. Across multiple trials, the experimental outcomes of the DeBERTa model consistently surpassed those of RoBERTa, underscoring the robust performance of the DeBERTa model.

In the final stage of the competition, a secret test dataset was introduced to assess the models' generalization performance. The experimental outcomes are presented in Table 5. We used a total of 19 models for voting, including 9-fold DeBERTa and 10-fold DeBERTa models, and selected the class with the highest frequency as the final result. The final F1_mean score was 0.8383. Using a combination of 9-fold DeBERTa, 10-fold DeBERTa, and 10-fold RoBERTa models, we used a total of 29 models for voting, and the final F1_mean was 0.8413. By further incorporating 10-fold XLM-RoBERTa models alongside the previous ones, totaling 39 models for voting, the final F1_mean was 0.8411. It can be observed that the fusion of different types of models is beneficial to the results. Although there was a slight decrease on the XLM-RoBERTa model, the diverse feature extraction capabilities among multiple models contribute significantly to the improvement of results.

6 Conclusion

In this paper, we have presented a comprehensive approach for text classification tasks on small-scale peer review datasets. By combining attention pooling, max pooling, and adversarial training (FGM), we achieved significant performance improvements. Through experimental validation, we have demonstrated the superiority of our method on small datasets. In the evolving era of deep learning, our approach amalgamates various techniques, providing an effective solution for text classification on small datasets. It overcomes the challenges posed by data scarcity, enhancing both model performance and robustness, offering novel insights and methodologies for addressing text classification challenges on small datasets.

	f1_mean	f1_case	f1_diso	f1_iscb	f1_rpkg	f1_scip	f1_secret
submission1	0.8383	0.829	0.842	0.836	0.854	0.889	0.779
submission2	0.8413	0.829	0.841	0.828	0.860	0.890	0.801
submission3	0.8411	0.831	0.847	0.828	0.860	0.882	0.798

Table 5: Final leaderboard scores for our submission

References

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Nils Dycke, Ilia Kuznetsov, and Iryna Gurevych. 2022. Nlpeer: A unified resource for the computational study of peer review. *arXiv preprint arXiv:2211.06651*.

Nils Dycke, Ilia Kuznetsov, and Iryna Gurevych. 2023. Argmining 2023 shared task - pragtag: Low-resource multi-domain pragmatic tagging of peer reviews. In *Proceedings of the 10th Workshop on Argument Mining*, Singapore. Association for Computational Linguistics.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Michael I Jordan. 1997. Serial order: A parallel distributed processing approach. In *Advances in psychology*, volume 121, pages 471–495. Elsevier.

Ilia Kuznetsov, Jan Buchmann, Max Eichler, and Iryna Gurevych. 2022. Revise and resubmit: An intertextual model of text-based collaboration in peer review. *Computational Linguistics*, 48(4):949–986.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. In *International Conference on Learning Representations*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.