Construction of English Resume Corpus and Test with Pre-trained Language Models

Chengguang Gan[1] Tatsunori Mori[1]
[1]Graduate School of Environment and Information Sciences
Yokohama National University
gan-chengguang-pw@ynu.jp tmori@ynu.ac.jp

Abstract

Information extraction (IE) is an essential NLP task, especially for extracting information from resumes. This study aims to transform the information extraction task of resumes into a simple sentence classification task, using an English resume dataset and improving the classification rules to create a larger and more fine-grained classification dataset. The new dataset is used to test the performance of mainstream pre-training language models, and experiments are also conducted to compare the impact of different training set sizes on the accuracy of the resume dataset. The results show that the improved annotation rules and increased sample size of the dataset improve the accuracy of the original resume dataset.

1 Introduction

As artificial intelligence develops, using artificial intelligence instead of HR for resume screening has always been the focus of research. And the accuracy of resume screening depends on the precision of resume information extraction.Hence, it is crucial to improve the precision of resume extraction for the subsequent steps of various analyses performance of resumes. The previous study on resume information extraction tends to use the Bi-LSTM-CRF model for Name Entity Recognition(NER) of resume text[1].Although this method extracts the resume information (e.g. Personal information, Name, Address, Gender, Birth) with high accuracy, it also loses some original verbal expression information. For example, the description of one's future career goals, requires complete sentences that cannot be extracted by the NER method. As an AI system that scores the candidate's resume, the career object is also part of the score. In summary, sentences such as these should not be ignored. Hence, in the prior study, the task of resume information extraction is transformed into a sentence classification task. Firstly, the various resume formats were converted into a uniform txt document. Then the sentences were classified after dividing them by sentence units. The classified sentences are used in the subsequent AI scoring system for resumes.The pilot study segmented and annotated 500 of the 15,000 original CVs from Kaggle.[1)] Five categories of tags were set: **experience**, **knowledge**, **education**, **project** and **others**[2)].The pilot study annotated resume dataset has problems, such as unclear classification label boundaries and fewer categories. Also, a dataset of 500 resumes with a total of 40,000 sentences in the tagging is sufficient for PLMs to fine-tune. If the dataset sample is increased, can the model's performance continue to improve.

To resolve all these problems, we improved the classification labels of resumes and used them to label a new resume classification dataset. To find out how many training samples can satisfy the fine-tune requirement of PLMs, we annotated 1000 resumes with a total of 78000 sentences. Furthermore, various experiments have been performed on the newly created resume dataset using the current mainstream PLMs.

2 Related Work

Since the last century, resume information extraction has been a critical applied research subfield in IE. In earlier studies, methods such as rule-based and dictionary matching were used to extract specific information from resumes[2]. HMM, and SVM methods extract information

1) https://www.kaggle.com/datasets/oo7kartik/resume-text-batch
2) https://www.kaggle.com/datasets/chingkuangkam/resume-text-classification-dataset

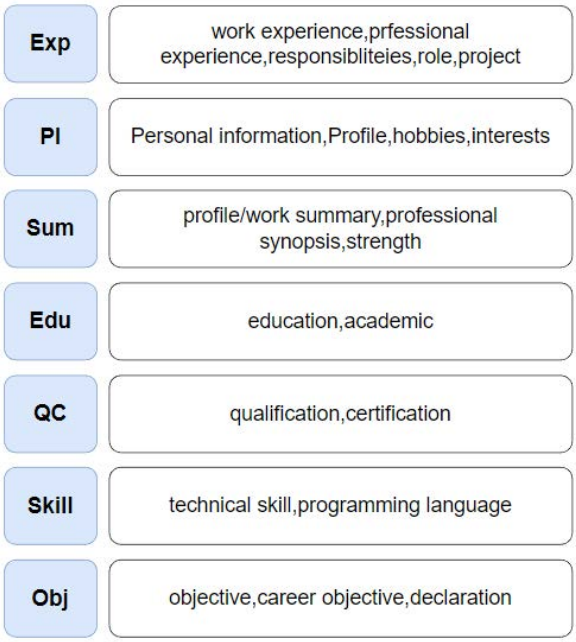

Figure 1: Resume annotation rules diagram.

such as a person's name and phone number from resume information[3].Related Resume Corpus Construct study has an extensive resume corpus in Chinese[4].

3 Corpus Construction

3.1 Annotation Rule

We increased the number of categories from 5 to 7 in order to discriminate the various parts of the resume more carefully.As shown in Figure 1, the blue block on the left is the abbreviation of the seven classification labels, and on the right is the name of the resume section corresponding to the label.The full names of the seven labels are **Experience**, **Personal Information**, **Summary**, **Education**, **Qualifications**, **Skill**, and **Object**.The newly developed classification rules make it possible to have a clear attribution for each item in the resume. It will not cause the neglect of some sentences in the resume, as there are **other** labels in the prior study.

3.2 Annotation Tool

In order to label resume datasets faster and more accurately, we developed a simple annotation program based on Tkinter[3)] .The operation interface of the resume annotation tool.This tool automatically recognizes original resumes in pdf, Docx, and txt formats. It can also segment all the sentences in the original resume according to a simple rule-based approach.Figure 2 shows the sample interface of the resume annotation tool. On the left are the sentences split by rule-based, and on the right are seven buttons that can be selected individually. After the sentence annotation of a whole resume is completed, a separate txt file will be automatically exported after closing the window, and the sentence annotation window for the next resume will be started automatically.

3) https://docs.python.org/ja/3/library/tkinter.html

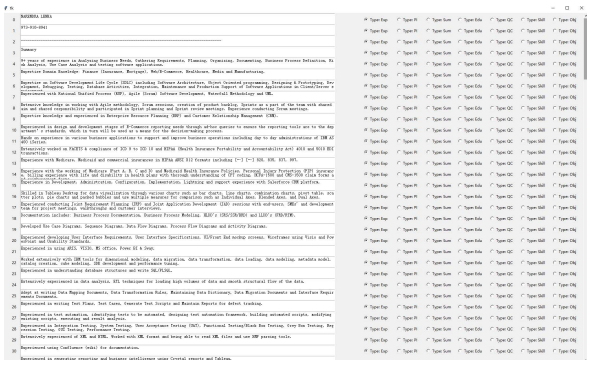

Figure 2: The operation interface of the resume annotation tool.

4 Experiments Set

In this section, we will perform various test experiments on the new-constructed resume dataset. First, we compared the performance of the BERT[5] model on the original resume corpus and the newly constructed resume dataset. Furthermore, four mainstream PLMs models are selected to test the resume dataset performance: BERT,ALBERT[6],RoBERTa[7], and T5[8]. For the fairness of the experiment, the size with the most similar parameters was chosen for each of the four models($\text{BERT}_{\text{large}}$, $\text{ALBERT}_{\text{xxlarge}}$, $\text{RoBERTa}_{\text{large}}$, T5_{large}).The evaluation metrics for all experiments were F1-micro.The training set, validation set, and test set are randomly divided in the ratio of 7:1.5:1.5.And each experiment was performed three times to take the average of the results.

5 Result

5.1 Pre-train Models Test

As shown in Table 2, the new resume corpus ameliorated by 0.35% over the original dataset F1-score for the same BERT model. RoBERTa and T5 scores improved by 0.68%

Sample	10000	15000	20000	25000	30000	35000	40000	45000	50000	55000
Valid	83	83.7	84.9	84.9	85.6	86	86.1	**86.6**	85.9	85.9
Test	83.5	84.3	85.3	85.6	84.6	85.4	**85.9**	**85.9**	85.8	85.1

Table 1: The first row indicates the number of training sets. The following two rows indicate the F1-score of the validation set and test set corresponding to the number of training samples.

Model	**F1-score**
BERT*$_{large}$(baseline)	86.32
BERT$_{large}$	**86.67**
ALBERT$_{large}$	**86.40**
RoBERTa$_{large}$	**87.00**
T5$_{large}$	**87.35**

Table 2: The first column * show accuracy of resume dataset before improvement.

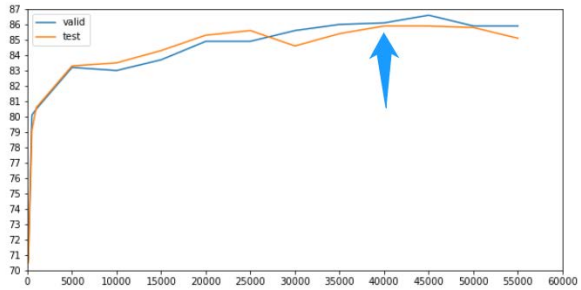

Figure 3: F1-score of different training samples.

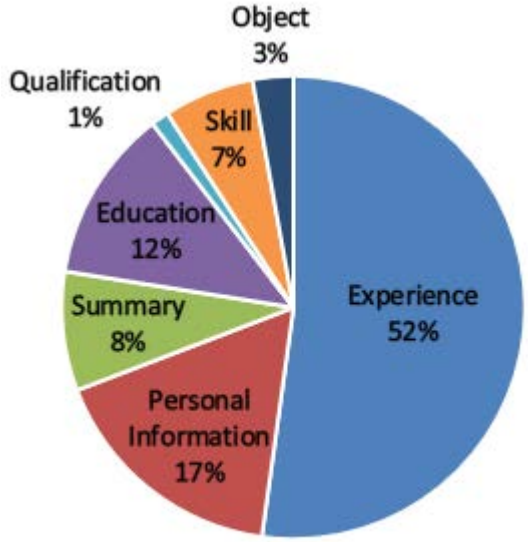

Figure 4: Fan chart of the percentage of each category of the resume corpus.

and 0.97% over baseline, respectively.The above results are also consistent with the ranking of the four PLMs in terms of their performance in various benchmark tests of NLP.

5.2 Sample Size Affects Experiment

In order to find out how many samples can bring out the maximum performance of the model, we divide the data set into training set 58000: validation set 10000: test set 10000. As shown in Table 1, the scores of the validation and test sets for different sample sizes. The model scores are tested from the 58000 training set, starting from 5000 and increasing the number of training samples every 5000.The highest score in the validation set is 86.6 when the training sample equals 45000. the highest score in the test set is 85.9 when the training sample equals 40000 and 45000.In order to visualize the relationship between the number of training samples and performance, we plotted the graphs (As Figure 3).It can be seen that as the number of training samples increases, the correctness of the model rises. Finally, the model's performance reaches the highest point when the training samples are increased to 40,000. From the experimental results, for the PLMs, this resume corpus above 40,000 is sufficient for the model's maximum performance.The results also prove that the new resume corpus, which doubles the sample size, is significant compared to the original resum corpus.

6 Analysis

In the final section, we analyze the sample distribution of the constructed resume corpus.Figure 4 shows that the category with the most significant proportion in the resume corpus is **experience**, which accounts for half of the resume text. In addition, the three categories that account for the least in the resume corpus are **skill**, **object**, and **qualification**, which account for only 7%, 3%, and 1%.Conclusively, resume text is a very easy sample imbalance for experimental subjects. Thus, the resume corpus also vigorously tests the model's learning capability for categories

with sparse samples in the training dataset. Hence, we plotted the conflation matrix of RoBERTa and T5 models. It is used to analyze the learning ability of the two models for sample-sparse categories in the dataset.

As shown in the figure 5, we can see the confusion matrix of RoBERTa and T5 models. First,the RoBE-RTa model is better for classifying **qualification** with the least number of samples. Secondly, the T5 model is slightly better than the RoBERTa model in terms of overall category classification results.The above results also demonstrate that our constructed resume corpus is highly unbalanced. However, if the model has strong performance, it can still learn the features of the corresponding category from very few samples.

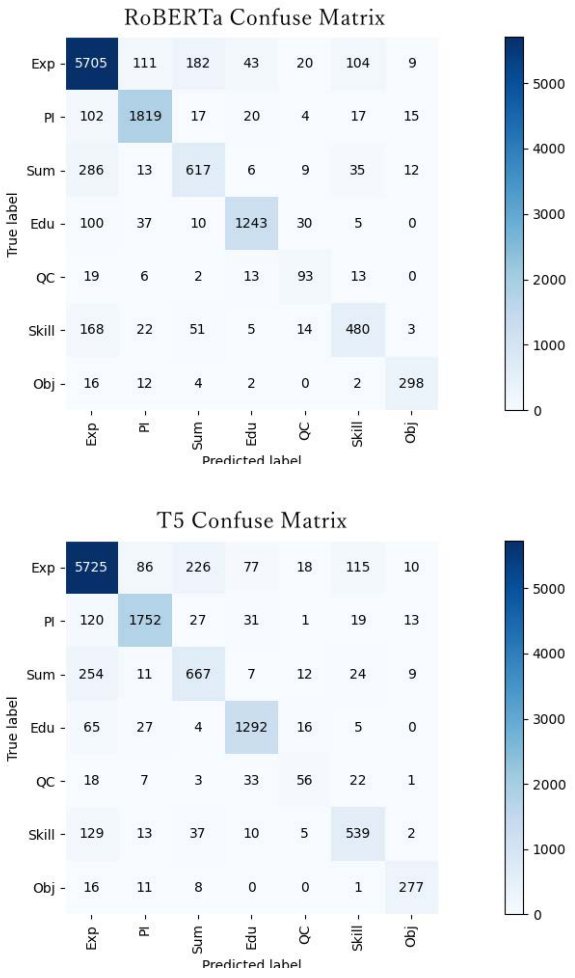

Figure 5: Confuse Matrix of $\text{RoBERTa}_{\text{large}}$ and T5_{large} model in test set.

7 Conclusion

In this paper, we improve the classification labels of the original English resume corpus. Furthermore, it doubled the number of samples size. The final tests and analyses also show the reliability of the newly constructed resume corpus. In future work, we will explore how to solve the sample imbalance problem of the resume corpus. Make the model learn effectively even for small sample categories.

References

[1] S. Huang, L. I. Wei, and J. Zhang. Entity extraction method of resume information based on deep learning. **Computer Engineering and Design**, 2018.

[2] R Mooney. Relational learning of pattern-match rules for information extraction. In **Proceedings of the sixteenth national conference on artificial intelligence**, volume 328, page 334, 1999.

[3] Kun Yu, Gang Guan, and Ming Zhou. Resume information extraction with cascaded hybrid model. In **Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL' 05)**, pages 499–506, 2005.

[4] Yanyuan Su, Jian Zhang, and Jianhao Lu. The resume corpus: A large dataset for research in information extraction systems. In **2019 15th International Conference on Computational Intelligence and Security (CIS)**, pages 375–378. IEEE, 2019.

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

[6] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. **arXiv preprint arXiv:1909.11942**, 2019.

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

[8] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. **J. Mach. Learn. Res.**, 21(140):1–67, 2020.