Lee at SemEval-2020 Task 5: ALBERT model based on the maximum ensemble strategy and different data sampling methods for detecting counterfactual statements

Junyi Li, Yuhang Wu, Bin Wang, Haiyan Ding*
School of Information Science and Engineering
Yunnan University, Kunming 65091, P.R. China
*Corresponding author, hyding@ynu.edu.cn

Abstract

This article describes the system submitted to SemEval 2020 Task 5: Modelling Causal Reasoning in Language: Detecting Counterfactuals. In this task, we only participate in the subtask A which is detecting counterfactual statements. In order to solve this sub-task, first of all, because of the problem of data balance, we use the undersampling and oversampling methods to process the data set. Second, we used the ALBERT model and the maximum ensemble method based on the ALBERT model. Our methods achieved a F1 score of 0.85 in subtask A.

1 Introduction

Now, with the rapid development of Internet technology, our society has entered a highly information era. Life is full of different kinds of information. However, the messy information is likely to change or even become the wrong information during the non-stop transmission process. This will have some bad effects. In order to solve this problem, we have created some positions where the verification information is right or wrong. However, this will consume a lot of labor costs and cause a waste of resources. At the same time, there are usually some misjudgments in human detection. Therefore, we need some new methods to deal. In the past two years, with the rapid development of artificial intelligence, some related problems have emerged in the field of deep learning. For example, common sense discrimination and natural language inference. The emergence of these problems provides some mechanical methods to solve the problem of error messages. Our research is also carried out.

Semeval 2020 task 5 (Yang et al., 2020) is to model causal inference in language: detect counterfactuals. This task is through the establishment of a model, and at the same time to detect whether it is a factual statement based on a large amount of text data. The generation of this task can help us to make a machine judgment of information correctness to a great extent. Subtask A is to detect counterfactual statements. This subtask requires us to determine whether a given statement is counterfactual. A counterfactual statement describes an event that has not actually occurred or cannot occur, and the possible consequences of the event. More specifically, counterfactuals describe events that are contrary to facts, and therefore naturally contain common sense, knowledge, and reasoning. Solving this problem is the basis of using natural language for all downstream counterfactual causal reasoning analysis. Subtask B is to detect precedents and consequences. This subtask states that causal insights are inherent features of counterfactuals. For this task, we mainly participated in subtask A.

In this task, we only participate in subtask A: Detect counterfactual statements. For this task, we use a combination of research and deep learning methods to deal with related tasks. In the task, we mainly use the lightweight Albert model based on the Transform mechanism. According to the latest progress of related research, the ALBERT model jointly researched by Google Research Center and Chicago Toyota Research Center has become our preferred model. Because it has fewer parameters and better performance than the BERT model. During data preprocessing, we found that the data set of subtask A has serious data imbalance problems. In order to solve the problem of data imbalance, we used undersampling and oversampling methods for comparative experiments. After data processing, we input the generated

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 664–669
Barcelona, Spain (Online), December 12, 2020.

word vector into the pre-trained model. In the model, we also adopted the method of maximum voting integration to optimize the performance of our model. In this task, our method is an effective way to get good performance.

The rest of our paper is structured as follows. Section 2 introduces related work. Section 3 describes data preparation. Methods are described in Section 4. Experiments and evaluation are described in Section 5. The conclusions are drawn in Section 6.

2 Related Work

In the field of natural language processing, common sense problems and natural inference problems are relevant to our task. These two problems have large data sets. By describing the data set, we can show the development of these two fields from the content of the data set. Meanwhile, the enlightenment of related solutions can also be obtained.

In the field of common sense, this question has some data sets. Event2Mind (Rashkin et al., 2018) is a crowdsourced corpus of 25,000 event phrases covering a diverse range of everyday events and situations. Situations with Adversarial Generations (SWAG) (Zellers et al., 2018) is a dataset consisting of 113k multiple choice questions about a rich spectrum of grounded situations. The winograd schema challenge (Trinh and Le, 2018) is a dataset for common sense reasoning. It employs winograd Schema questions that require the resolution of anaphora: the system must identify the antecedent of an ambiguous pronoun in a statement. Reading Comprehension with Commonsense Reasoning Dataset (ReCoRD) (Zhang et al., 2018) is a large-scale reading comprehension dataset which requires commonsense reasoning.

In the field of natural inference, there are also a large number of data sets. Such as, The Stanford Natural Language Inference (SNLI) (Bowman et al., 2015) Corpus contains around 550k hypothesis/premise pairs. The Multi-Genre Natural Language Inference (MultiNLI) (Williams et al., 2017) corpus contains around 433k hypothesis/premise pairs. It is similar to the SNLI corpus, but covers a range of genres of spoken and written text and supports crossgenre evaluation. The SciTail (Khot et al., 2018) entailment dataset consists of 27k. In contrast to the SNLI and MultiNLI, it was not crowd-sourced but created from sentences that already exist in the wild.

3 Data Preparation

In this part, we mainly introduce our processing of the data set. At the same time, for the problem of data imbalance, we will also introduce the methods of undersampling and oversampling. Through two methods, we will solve the problem of data imbalance.

3.1 Data processing

The organizers provided train and test sets, containing 13000 and 7000 data respectively. Cleaning the text before further processing helps to generate better functionality and semantics. We perform the following preprocessing steps.

- We know that some repeated symbols have no meaning. As a result, repeated periods, question marks and exclamation marks are replaced with a single instance with the special mark ”repeat” added.
- All contractions were changed to complete parts. This helps the machine understand the meaning of words (for example:”there’re” changed to ”there” and ”are”.
- Twitter data contains a lot of emojis. Emojis can cause the number of unknown words to rise, which can lead to poor pre-training effects. Emoticons (for example, ”:(”, ”:” ”,”: P ”and emoticons, etc.) are replaced by emotional words with their own meaning. This will improve the pre-training effect.
- Generally, words have different forms according to the change of context. However, different forms of words will cause ambiguity in pre-training and affect the effect of pre-training. Lexicalization, through WordNetLemmatizer to restore language vocabulary to the general form (can express complete semantics).
- Tokens are converted to lower case.

3.2 Undersampling and Oversampling

In this task, by counting the number of label categories in the train set, we found that the label of the data has an imbalance problem. The number of label categories in our train data set is shown in Figure 1.

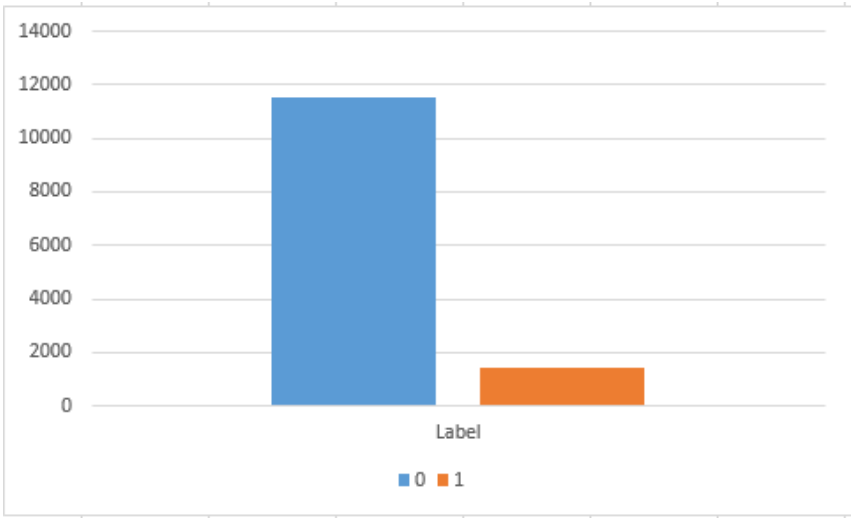

Figure 1: The number of label categories in train datasets

In Figure 1, we clearly see that the data set has a very unbalanced problem. These problems will reduce the performance of our model. Therefore, we need to solve the problem of data imbalance. In order to solve the problem of data imbalance, we mainly adopted the methods of undersampling and oversampling. The two methods are as follows.

- Undersampling is to sample only a part of the samples from the major category, and select as many samples as the minority category. This sampling maintains the original data distribution of the category. We only need to use fewer samples to make the data balanced. In our task, we sample the data of label '0' to make it the same as the data of label '1'.
- Oversampling is that we copy the samples in a few categories, so that the number is as large as most samples. The copy operation needs to maintain the original data distribution of a few samples. We don't need to get more data to make the data set balanced. The sampling method is a good method for class balance. In our task, we copy the data of the label 1 as much as the data of the label 0.

Our data set is processed by these two methods. The label distribution is shown in Table 1.

methods	label 0	label 1	all
original	11546	1454	13000
undersampling	1546	1454	3000
oversampling	11546	11632	23178

Table 1: The label distribution in the train datasets.

4 Methods

In this part, we mainly introduce the ALBERT model and the maximum ensemble method based on the ALBERT model.

4.1 ALBERT

Google company introduces a new language representation model called bert, which stands for Bidirectional Encoder Representations from Transformers. Unlike other language representation models, BERT

is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers.

However, the large-scale parameters of the BERT model cause the exponential growth of training time and the shortage of computing resources. At the same time, too many parameter scales will cause the performance of the model to decline. Therefore, Google and Toyota technological institute have proposed a new model, which is the ALBERT model (Lan et al., 2019).

ALBERT incorporates two parameter reduction techniques that lift the major obstacles in scaling pre-trained models. The first one is a factorized embedding parameterization. By decomposing the large vocabulary embedding matrix into two small matrices, we separate the size of the hidden layers from the size of vocabulary embedding. This separation makes it easier to grow the hidden size without significantly increasing the parameter size of the vocabulary embeddings. The second technique is cross-layer parameter sharing. This technique prevents the parameter from growing with the depth of the network. Meanwhile, a self-supervised sentence order prediction (SOP) loss is also introduced. SOP focuses on the coherence between sentences, and aims to solve the invalidity of the sentence prediction loss.

In this task, we used the ALBERT model to get good performance. Our model is shown in Figure 2.

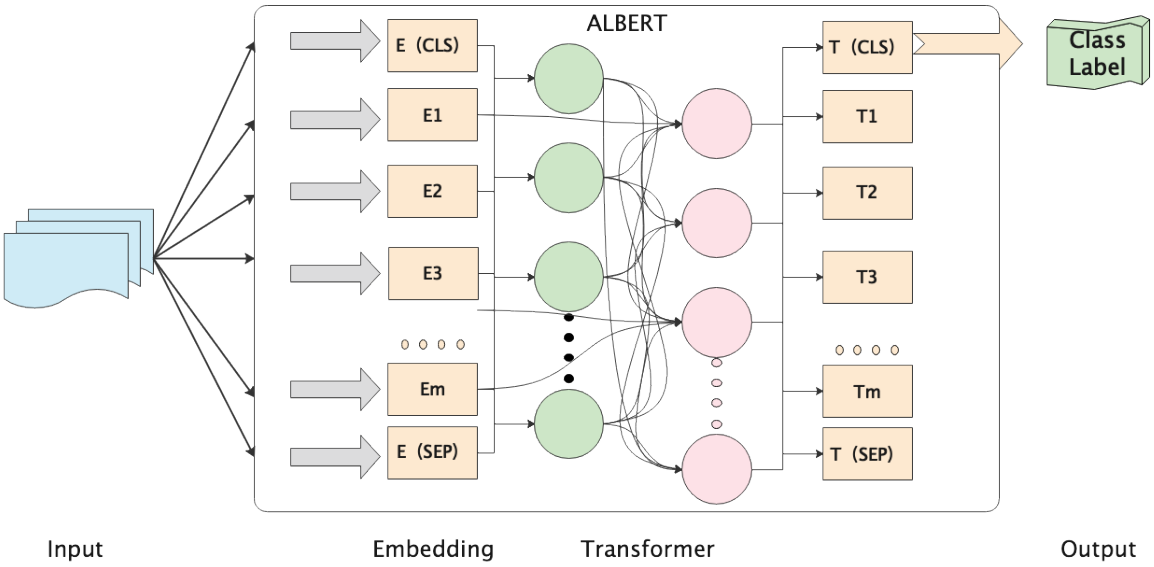

Figure 2: The architecture of the ALBERT model in our task

where E represents the embeddings of ALBERT and T represents the process of transformer mechanism. Meanwhile, the E[CLS] and E[SEP] are added at the beginning and end of each instance, respectively.

4.2 Max ensemble

We know that common ensemble methods are based on voting ensemble (Wang et al., 2018). In this paper, we first fine-tune the ALBERT model with different random seeds. For each input, we will output the best predictions and probabilities made by fine-tuned ALBERT, and summarize the prediction probabilities of each model. The output of the integrated model is the prediction with the highest probability. We call this integration method voting ensemble, and the ALBERT model after voting ensemble (Xu et al., 2020) has been significantly improved.

In our task. Our voting ensemble is chosen when outputting the maximum performance. Our voting ensemble is shown in Figure 3,where x represents our input and y represents the model integrated output.

5 Experiments and evaluation

In this task, we use the ALBERT model to train the task. For the ALBERT model, the main hyper-parameters we focused on are the training step size, batch size, warm steps, and learning rate. After

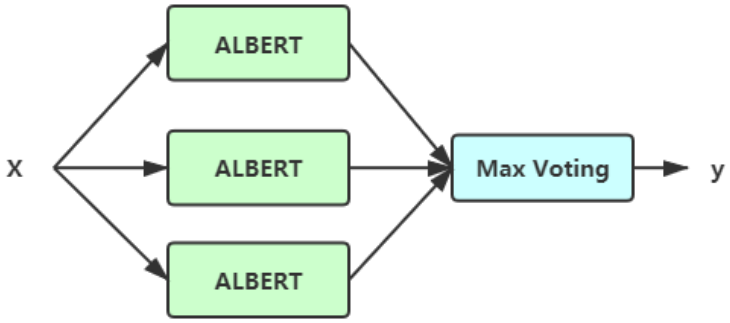

Figure 3: The architecture of max ensemble in our task

learning the hyper-parameter adjustment for similar tasks, we fine-tune the model hyper-parameters. As show in Table 2.

train step	learning rate	batch size	warm steps
23800	5e-6	32	1256

Table 2: Details of the hyper-parameters.

This task mainly uses precision, recall and F1 score for performance evaluation. In this task, we mainly use the ALBERT model for experiments. Meanwhile, in order to optimize the model performance, we also adopted different sampling methods and maximum ensemble methods. Our experimental results are shown in Table 3.

Model	Dataset	Precision	Recall	F1
Albert(single)	original	0.78	0.76	0.77
Albert(single)	undersampling	0.79	0.78	0.78
Albert(single)	oversampling	0.77	0.78	0.77
Albert(ensemble)	original	0.84	0.82	0.83
Albert(ensemble)	**undersampling**	**0.85**	**0.84**	**0.85**
Albert(ensemble)	oversampling	0.85	0.84	0.84

Table 3: Performance with our methods.

From this table, we can see that the undersampling data processing method and the maximum ensemble method can effectively optimize the effectiveness of our model. So, for this task, our method is an effective method.

6 Conclusion

In this task, we mainly use the ALBERT model. Based on the model, we also adopted the method of maximum ensemble. At the same time, due to the imbalance of the task data set, we also used the methods of undersampling and oversampling to deal with the problem of data imbalance. In this task, we found that using undersampling and maximum ensemble methods can achieve optimal performance. In the task, we did not think too much about why undersampling brought more improvements in results. In the future work, we will devote ourselves to finding the reason.

However, in the rankings, the performance of our model is still not the best. We analyze that the reason for this situation may be that the model parameter fine-tuning has not reached the best and the data imbalance processing method is not good enough.

In the future, we will improve our methods towards these reasons. We believe that we can get a good result. Meanwhile, in order to solve the problem more comprehensively, we will try to participate in subtask B.

Acknowledgements

This work was supported by the Natural Science Foundations of China under Grant 61463050, the Science Foundation of Yunnan Education Department under Grant 2019Y0005.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. *CoRR*, abs/1508.05326.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

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

Hannah Rashkin, Maarten Sap, Emily Allaway, Noah A Smith, and Yejin Choi. 2018. Event2mind: Commonsense inference on events, intents, and reactions. *arXiv preprint arXiv:1805.06939*.

Trieu H Trinh and Quoc V Le. 2018. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*.

Jin Wang, Bo Peng, and Xuejie Zhang. 2018. Using a stacked residual lstm model for sentiment intensity prediction. *Neurocomputing*, 322:93–101.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference. *CoRR*, abs/1704.05426.

Yige Xu, Xipeng Qiu, Ligao Zhou, and Xuanjing Huang. 2020. Improving bert fine-tuning via self-ensemble and self-distillation. *arXiv preprint arXiv:2002.10345*.

Xiaoyu Yang, Stephen Obadinma, Huasha Zhao, Qiong Zhang, Stan Matwin, and Xiaodan Zhu. 2020. SemEval-2020 task 5: Counterfactual recognition. In *Proceedings of the 14th International Workshop on Semantic Evaluation (SemEval-2020)*, Barcelona, Spain.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *arXiv preprint arXiv:1808.05326*.

Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018. Record: Bridging the gap between human and machine commonsense reading comprehension. *arXiv preprint arXiv:1810.12885*.