HW-TSC at SemEval-2022 Task 7: Ensemble Model Based on Pretrained Models for Identifying Plausible Clarifications

Xiaosong Qiao, Yinglu Li, Min Zhang, Minghan Wang,
Hao Yang, Shimin Tao, Ying Qin
Huawei Translation Services Center, Beijing, China
{qiaoxiaosong, liyinglu, zhangmin186, wangminghan,
yanghao30, taoshimin, qinying }@huawei.com

Abstract

This paper describes the system for the identifying Plausible Clarifications of Implicit and Underspecified Phrases. This task was set up as an English cloze task, in which clarifications are presented as possible fillers and systems have to score how well each filler plausibly fits in a given context. For this shared task, we propose our own solutions, including supervised approaches, unsupervised approaches with pretrained models, and then we use these models to build an ensemble model. Finally we get the 2nd best result in the subtask1 which is a classification task, and the 3rd best result in the subtask2 which is a regression task.

1 Introduction

The rapid development of artificial intelligence has also been reflected in the field of NLP, and there have been many heavyweight achievements, such as word2Vec (Mikolov, et al., 2013), Glove (Pennington, et al., 2014), Transformer (Vaswani, et al., 2017). Natural language processing is an important branch of artificial intelligence. Cloze tasks have become a standard framework for evaluating various discourse-level phenomena in NLP, which is an important field in artificial intelligence, many researchers have long been committed to the development of this field. Some prominent examples include the narrative cloze test (Chambers and Jurafksy, 2008), the story cloze test (Mostafazadeh et al., 2016), and the LAMBADA word prediction task (Paperno et al., 2016). Cloze requires the testee to infer from the context, which is very difficult for machines.

The goal of this shared task is to evaluate the ability of NLP systems to distinguish between plausible and implausible clarifications of an instruction. Such clarifications can be critical to ensure that instructions describe clearly enough what steps must be followed to achieve a specific goal. This task was set up as a cloze task. However, different from regular cloze task, there may be zero or more than one correct candidates out of the five options. This presents new challenges for cloze systems.

For subtask 1, it is a classification task that requires the system to classify five candidates into corresponding categories, which are plausible, neutral, or implausible, and the number of each category is not fixed. This means that there may be zero or more than one correct candidates out of the five options, and the same applies to the other two categories. This situation creates new challenges for cloze tasks.

For subtask 2, it is a regression task ask annotators to rate for each clarification option whether it "makes sense in the given how-to guide" (on a scale from 1 to 5) to assess the plausibility of different clarification options.

In this paper, we analyze the characteristics of the shared task and describe out contribution to this cloze task. We build an ensemble model with Deberta-v3 (He P, et al., 2020), Roberta-large (Liu Y, et al., 2019), SBERT (Reimers and Gurevych, 2019), including supervised approaches and unsupervised approaches. Our model had the 2nd best performance in the subtask1 (66.1% Accuracy Score) and the 3rd best performance in the subtask2 (77.4% Ranking Score). The results are encouraging for evaluating various discourse-level phenomena in NLP, although there is much room for improvement.

The rest of this paper is organized as follows. Section 2 introduce our approach for the shared task. Section 3 shows the experimental results of our approach and do some analysis. In Section 3, experimental results are compared and discussed. Finally, the whole paper is summarized with a brief conclusion in Section 4.

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1056 - 1061

2 System Overview

In this section, we first describe our data processing steps. We experimented with different ways of processing the data, trying to find the one that worked best for the task. And then, we discuss our solutions with pre-trained models for the shared task, including unsupervised approaches, supervised approaches, and an ensemble model.

2.1 Data Processing

The participants of the shared task were provided a collection of revisions of instructional texts from the how-to website wikiHow. The dataset contains sentences that need to be filled in and its previous context, follow-up context, five options, etc. The data example is as shown in the Figure 1 below:

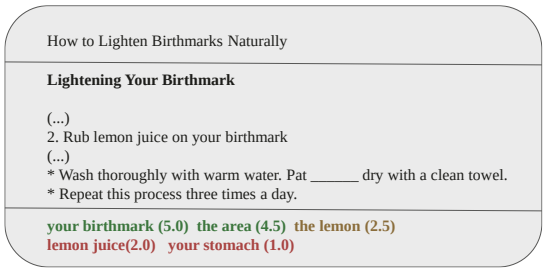

Figure 1: Data example

We bring the five options into the positions that need to be filled in, and get a dataset that is five times the size of the original.

For subtask1, a label file was gave which contains the corresponding category of each option of each piece of data, which category does it belong to, plausible, neutral or implausible? We map these three categories to numbers 2, 1, 0, corresponding to plausible, neutral and implausible.

For subtask2, a score file was gave to assess the plausibility of different clarification options. For the score file, we keep it in its native state. In addition, in order to make the model focus on the positions that need to be filled in, we have added special symbols $ on both sides of the blank. After data processing, the data example is as shown in the Figure 2 below:

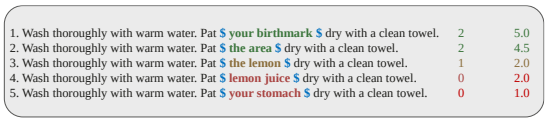

Figure 2: Data example after processing

To get more information that might be useful, we tried a variety of sentence concatenations using different columns in the data. Our experiments show that this is necessary and effective.

2.2 Unsupervised approach

First, in order to get a reliable benchmark on this task, we use unsupervised methods to try to solve the task with BERT. Because the pre-training process of BERT includes masked language model, that is, to replace a small part of words in the text with [MASK], and let the model predict the words replaced by [MASK]. This task is very similar to cloze, so we can use cloze to test BERT's masked language model capability in longer and more [MASK] texts (Ding et al., 2021).

For this task, we tokenize each option to get the number of tokens, and then fill in the blank with the same [MASK] as the number of tokens. We do this because multiple [MASK] work better than a single one. The processing process is shown in the following Figure 3:

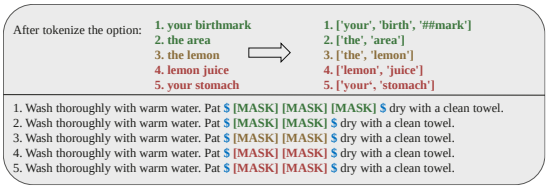

Figure 3: Data example after filled with [MASK]

A pooling operation is added to the output of BERT to generate a fixed-size sentence embedding vector. The tokens embedding of [MASK] obtained from the pre-trained model would be used for classification with different pooling strategies. In our experiment, three pooling strategies were used for comparison:

- MEAN strategy

  Calculate the average value of each token output vector of option to represent the sentence vector.
- MAX strategy

  Take the maximum value of each dimension of all output vectors of option to represent the sentence vector.
- SUM strategy

  Take the sum value of each dimension of all output vectors of option to represent the sentence vector.

2.3 Supervised approach

After getting a benchmark with an unsupervised method, we want to get some experimental results

with a supervised method. First, we still conduct some experiments to screen out the model with better performance from several models. The models we use include Deberta-v3, Roberta-large, SBERT, BERT (Devlin et al., 2018), etc. The final experimental results will be displayed in the experimental section.

And then, in the above part, we mentioned filling in the blanks with [MASK], and using the embedding of [MASK] is directly used for classification. This is naturally associated with the similarity between [MASK] and options. Intuitively, [MASK] should be the most similar to the plausible option, and the least similar to the implausible option. So we use SBERT to calculate the similarity between the sentences after filling in [MASK] and filling in the options. After getting the similarity, we classify it by threshold optimization. The schematic diagram of the data process is shown in the following Figure 4:

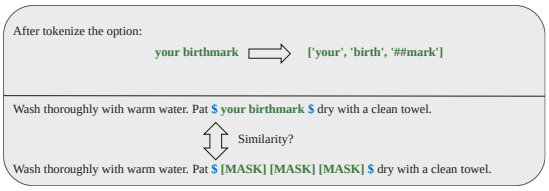

Figure 4: Data process for calculate

The sentence obtained by filling [MASK] into the blank part and the sentence obtained by filling the option into the blank part are used as the input of the model, and then the embedding representation of [MASK] and the option is obtained by average pooling, as u and v respectively. We concatenate the values of u and v and the absolute value of their differences for classification tasks, and we also calculate the similarity between u and v for the task. The process is shown in Figure 5.

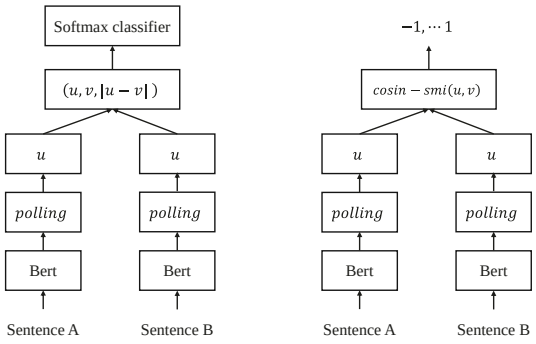

Figure 5: The process of SBERT

2.4 Model ensemble

Through Sections 2.3.1 and 2.3.2, we have obtained the results of several models. By comparing the classification results between different models, there are large differences, which means that for the classification results of the same data, the Model I may classify it into IMPLAUSE, but the Model II may classify it as NEUTRAL. This makes it possible for us to further improve the classification effect through the model ensemble.

The voting method is an ensemble learning model that follows the majority principle, and reduces variance through the integration of multiple models, thereby improving the robustness and generalization ability of the model. We adopt the voting method commonly used in ensemble learning, which is an ensemble learning model that follows the principle of majority rule by the minority, and reduces variance through the integration of multiple models, thereby improving the robustness and generalization ability of the model. We used four models (Roberta based on unsupervised method, and Roberta-large, Deberta-v3, SBERT based on supervised method) as benchmarks for ensemble learning. The structure of ensemble model is shown in the Figure 6.

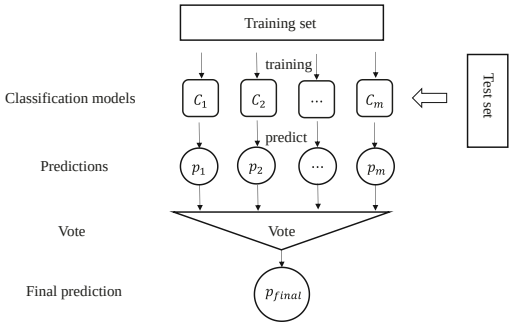

Figure 6: The structure of ensemble model

3 Experimental Results

In the following experimental part, all the data used for the experiment adopts the data processing method we introduced in Section 2.1.

3.1 Unsupervised approach results

We propose an attempt to use an unsupervised approach to benchmark this task in Section 2.2, and propose three strategies for dealing with [MASK]. The experimental comparison of the three strategies is gave by Table 1, there is little

Model	Pooling	Train Accuracy	Dev Accuracy
Bert-large	Mean	0.4373	**0.5152**
Bert-large	Sum	0.4434	0.4920
Bert-large	Max	0.4330	0.5150
Roberta-large	Mean	0.4678	**0.5788**

Table 1: Pooling strategy for unsupervised approach

Model	Train Accuracy	Dev Accuracy
Bert-base	0.5071	0.5394
Bert-large	0.5599	0.5613
Roberta-large	0.5260	0.5710
Deberta-v3	0.4403	**0.6326**

Table 2: Screen out the model with better performance from several models with [CLS] embedding

Model	Strategy	Train Accuracy	Dev Accuracy
Roberta-large	Classification	0.5463	0.5665
Roberta-large	Similarity	0.6337	**0.6272**
Bert-base	Classification	0.6298	0.5237
Bert-base	Similarity	0.7034	0.4553

Table 3: Experiment results of SBERT

Model	Train Accuracy	Dev Accuracy	Test Accuracy
Roberta-large unsupervised	0.4678	0.5788	--
Roberta-large supervised	0.5260	0.5710	--
Deberta-v3	0.5624	0.6485	0.622
SBERT(Roberta-large)	0.6337	0.6272	--
Ensemble	--	**0.7088**	**0.661**

Table 4: Results for subtask1 with ensemble model

Model	Train Rank	Dev Rank	Test Rank
Roberta-large unsupervised	--	0.6112	--
Roberta-large supervised	--	0.6370	--
Deberta-v3	0.6137	**0.7784**	0.747
SBERT(Roberta-large)	--	0.6560	--
Ensemble	--	0.7752	**0.774**

Table 5: Results for subtask2 with ensemble model

difference between MEAN strategy and Max strategy. We ended up using MEAN strategy to get a benchmark (57.88% Accuracy Score) with Roberta-large.

3.2 Supervised approach results

For supervised methods, although we did some experiments to try to find a better embedding than [CLS] for this task, we didn't get it. So we still ended up screening out the model with better performance from several models with [CLS] embedding. The results of model screening are given in Table 2.

In Section 2.3 we propose to use SBERT to try to solve this task. We conduct experiments with direct classification and computing similarity respectively. Table 3 gave the experimental results of SEBRT.

3.3 Model ensemble results

After obtaining several benchmarks using the unsupervised method and the supervised method,

respectively, in the supervised method, by adjusting the parameters of several models, such as adjusting the batch size, learning rate or freezing some parameters in the model. We end up with 11 results including the above for ensemble learning。Due to space limitations, we no longer list the training results after adjusting the model parameters or freezing some parameters here. For each model, we list its best results. The results are given in Table 4.

For subtask 2, we just converted the above model from a classification task to a regression task, and also adjusted the training parameters, froze some model parameters, and obtained eleven kinds of results. The ensemble learning is carried out by the method of averaging, and the final result is obtained and shown in Table 5.

3.4 Discussion

A phenomenon can be observed from the experimental results: when the model has not fully converged on the training set, the best result of the model on the validation set has already appeared. Especially when using Deberta-v3, when the best results (63.26%) appear on the validation set, the model's accuracy score on the training set is only 44.03% in the classification task. This phenomenon also occurs in the regression task. But the difference is that the difference between the results of the training set and the validation set of the model in the classification task is much smaller than that in the regression task.

We therefore consider that there is noise in the training set, which is especially evident in classification tasks. To verify that there is really noise in the data, we compared part of the data in the training set with the data on the wikiHow website, as shown in the figure below. It can be seen that the sentences that appear in the original text in time are still marked as Neutral or Implausible in the training set. Figure 7 shows an example of original data with id-20 that may be incorrect.

Add in a few drops of clear nail polish and stir with a toothpick until there are no lumps. Keep stirring until you get an even color and consistency. If the color too sheer, add some more eyeshadow. ***Make sure that there are no clumps in the polish.*** *If there are any clumps, break them up with the toothpick. If you don't do this, they will show up on your manicure and make it look lumpy.*

Figure 7: Data in the original paragraph

We tried Label Smoothing (Müller et al., 2019) and Self-Adaptive Training (Huang et al., 2020) to solve the problem of data noise. Although there is no significant improvement in the model's performance, it speeds up the model Convergence rate during training.

4 Conclusion

In this paper, we describe the Identifying Plausible Clarifications of Implicit and Underspecified Phrases shared task held within SemEval-2022 and present the design, the data, the results, and the systems for the shared task. The participants of the shared task were provided a collection of revisions of instructional texts from the how-to website wikiHow. The shared task is challenging, partly due to the relatively small training data and label noise.

We develop an ensemble model of NLP to distinguish between plausible and implausible clarifications of an instruction, achieving the 2nd best performance in the subtask1 and the 3rd best performance in the subtask2. For some of the problems reflected in this task, such as data noise, non-identically distributed data, there is still a lot of research space.

References

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised learning of narrative event chains. In *ACL 2008, Proceedings of the 46th Annual Meeting*

*of **the Association for Computational Linguistics, June** 15-20, 2008, Columbus, Ohio, USA*, pages 789–797. The Association for Computer Linguistics.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James F. Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In NAACL HLT 2016, *The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 839–849. The Association for Computational Linguistics.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. 2016. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7*, 2021. OpenReview.net.

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. CoRR, abs/1907.11692.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Minjie Ding, Mingang Chen, Wenjie Chen, and Lizhi Cai. 2021. English cloze test based on BERT. In *Knowledge Science, Engineering and Management -14th International Conference, KSEM 2021, Tokyo, Japan, August 14-16, 2021, Proceedings, Part II, volume 12816 of Lecture Notes in Computer Science*, pages 41–51. Springer.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Rafael Müller, Simon Kornblith, and Geoffrey E. Hinton. 2019. When does label smoothing help? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 4696–4705.

Lang Huang, Chao Zhang, and Hongyang Zhang. 2020. Self-adaptive training: beyond empirical risk minimization. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.*

Michael Roth, Talita Anthonio, and Anna Sauer. SemEval-2022 Task 7: Identifying plausible clarifications of implicit and underspecified phrases in instructional texts. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*.