hub at SemEval-2021 Task 2: Word Meaning Similarity Prediction Model Based on RoBERTa and Word Frequency

Bo Huang, Yang Bai, Xiaobing Zhou*
School of Information Science and Engineering
Yunnan University, Yunnan, P.R. China
⋆Corresponding author:zhouxb@ynu.edu.com

Abstract

This paper introduces the system description of the hub team, which explains the related work and experimental results of our team's participation in SemEval 2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation (MCL-WiC). The data of this shared task is mainly some cross-language or multi-language sentence pair corpus. The languages covered in the corpus include English, Chinese, French, Russian, and Arabic. The task goal is to judge whether the same words in these sentence pairs have the same meaning in the sentence. This can be seen as a task of binary classification of sentence pairs. What we need to do is to use our method to determine as accurately as possible the meaning of the words in a sentence pair are the same or different. The model used by our team is mainly composed of RoBERTa and Tf-Idf algorithms. The result evaluation index of task submission is the F1 score. We only participated in the English language task. The final score of the test set prediction results submitted by our team was 84.60.

1 Introduction and Background

With the continuous development of science and technology, we are now in an era of massive data. We cannot use manual methods in the processing and retrieval of text data. Especially in the work of comparing and calculating the semantic difference at the word level in the text. In this type of work, automatic processing of text data with machines has become a new choice. The research on the detection method (Resnik, 1995; Miller and Charles, 1991) and evaluation method (Sánchez et al., 2012) of semantic similarity has become a subject of wide discussion. Specific application scenarios have been produced in some fields of natural language processing and information retrieval. Such as sentiment analysis (Araque et al., 2019), medical disease similarity query (Mathur and Dinakarpandian, 2012), text question and answer(Mohler and Mihalcea, 2009) etc.

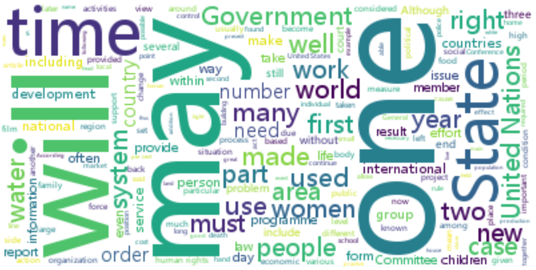

Figure 1: A word cloud diagram of the training set text data provided by the task organizer team. The result shown in the figure is the data after removing the stop words.

Similar to humans' strategies for detecting the meaning of words in different sentences, machines and algorithms also need to predict the results based on the context. Therefore, the method of generating vectors based on each word is not suitable for such tasks. For example Word2Vec (Mikolov et al., 2013). Based on the characteristics of text serialization, extracting contextual information in the text as the input of the model will provide the model with richer and more accurate information. For example, in dealing with the problem of polysemous and synonymous words. The ELMo (Peters et al., 2018) method based on LSTM (Shi et al., 2015) overcomes the difficulty that the model cannot learn the context. ELMO can dynamically adjust word embedding according to the context, so it can solve the problem of ambiguity. However, the use of a bidirectional LSTM as a feature extractor makes its training time and feature extraction effect unsatisfactory. In the follow-up work, the appearance of Transformer (Vaswani et al., 2017) introduces new and better feature extractors for the model. The BERT (Devlin et al., 2019) model based on Trans-

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 719–723
Bangkok, Thailand (online), August 5–6, 2021. ©2021 Association for Computational Linguistics

long
Government
right
women
work
many
one
two
time
year
member
will
may
area
used
new
people
children
use
well
made
first
system
provide
use
made
two
time
work
used
one
first
development
year
many
new
will
Government
State
women

(a) The validation set data

(b) The test set data

Figure 2: The word cloud diagram of the validation set and test set data provided by the task organizer team. The result shown in the figure is the data after removing the stop words.

former Encoder (Vaswani et al., 2017) achieved the best results in many NLP tasks.

We participated in SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation (MCL-WiC) English task. This task is to predict whether a word with the same part of speech has the same meaning in a sentence pair (Martelli et al., 2021). We are inspired by the work of Chen, Weilong and others on the task of predicting the influence of context on word similarity (Chen et al., 2020), and use methods based on RoBERTa (Liu et al., 2019) and Tf-Idf (Ramos et al., 2003) to complete the task. At the same time, we also tried to combine ALBERT (Lan et al., 2020) with BERT (Devlin et al., 2019) and Tf-Idf to observe their performance on the English data set. We introduce our methods and experiments in detail in Sections 2 and 3. Our model code can provide reference [1].

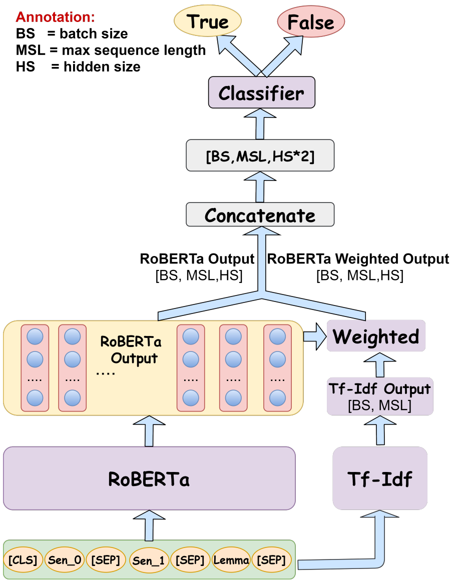

Figure 3: The model structure and data flow we used in the task.

2 Data and Methods

In this section, we will introduce the data we use in the task and the models and methods we use.

2.1 Data Description

The task organizer team provides each team with training data sets, validation data sets, and test data sets related to the ”Multilingual and Cross-lingual Word-in-Context Disambiguation” task. Because we only successfully submitted the test set prediction results of the English task, we only discuss the English data set here. The training data set and the validation data set are composed of two parts. The first part contains the ID, the lemma of the target word, the part of speech of the target word, the sentence pair data, and the position index of the target word in the sentence pair. The target word is usually only one word, and they have the same part of speech in the sentence pair. The second part is whether the target words appearing in the sentence pair are tags with the same meaning.

If two words have the same meaning, it is ”True”, otherwise it is ”False”. The sentence lengths in the sentence pairs are not the same. Compared with the training data set and the validation data set, the test set only contains the first part mentioned above. We need to use our method to predict whether the same words appearing in sentence pairs in the test set have the same meaning. Table 1 shows a sample of sentence pair data we used in the task.

There are 8000 and 1000 data in the training set and validation set respectively. The proportions of the “True” label and the “False” label in the training set and the validation set are the same, both are 50% and 50%. There are 1000 pieces of data in the test set. Information about word frequency

[1] https://github.com/Hub-Lucas/hub-at-task2

ID	Lemma	Part of Speech	Sentence	Start	End
151	excess	NOUN	We want to rebuild our country, which was dismantled by the **excesses** of Mobutu	60	68
151	excess	NOUN	More often than not, words per page are well in **excess** of that standard.	48	54

Table 1: The sample data of a pair of sentence pairs we use in the task.

$$[Tf - Idf\_Output]_i \times [RoBERTa\_Outpt]_i = [Weighted]_i \tag{1}$$

$$[Tf - Idf\_Output]_i^T \times [Weighted]_i = [RoBERTa\_Weighted\_Output]_i \tag{2}$$

$$0 \leq i < batch\_size \tag{3}$$

will be involved in our method. We use word cloud graphs to visualize the text data in the training set and the text data in the test set. The word cloud image clearly shows us the characteristics of word frequency distribution in the text data set. Figure 1 and Figure 2 show the word frequency information in the training set, validation set, and test set.

2.2 Methods

Combined with the analysis and understanding of task description and task data set, we chose to develop a system based on RoBERTa and Tf-Idf. Besides, we also tried to use the combination of ALBERT (Lan et al., 2020), BERT (Devlin et al., 2019) and Tf-Idf to verify their effect on the verification set. Due to the addition of the attention mechanism, Transformer has achieved good results in multitasking in the field of natural language. The three models of BERT, ALBERT, and RoBERTa are all based on the improvement of the transformer architecture. Compared with BERT, ALBERT not only has fewer parameters, but also has the characteristics of parameter sharing between different layers (Lan et al., 2020; Devlin et al., 2019). Therefore, ALBERT is better than BERT in terms of memory space and training time. Compared with ALBERT (Lan et al., 2020), RoBERTa (Liu et al., 2019) does not perform the task of predicting the next sentence during the pre-training process, and also uses a new dynamic masking mechanism. At the same time, the pre-training time of the RoBERTa model is longer, using a larger batch size, and the corpus data used for pre-training is also larger (Liu et al., 2019).

In our system, the first step is to use the preprocessed data as the input data of RoBERTa and Tf-Idf. In the second step, we get the output result of the last layer of RoBERTa (RoBERTa Output) and the output result of Tf-Idf (Tf-Idf Output). In the third step, we use the output result of Tf-Idf to weight the output result of RoBERTa. We can get a weighted result, we call it RoBERTa weighted output. In the fourth step, we connect the RoBERTa output result and the RoBERTa weighted output result together. In the fifth step, we use the result of the previous step as the input of the classifier. Use the classifier to output the prediction results of the model. In the final step, the results of the model prediction are processed into the format required by the task organizer team.

Among them, the shape of RoBERTa output [batch_size, max_sequence_length, hidden _size]. The shape of Tf-Idf output is [batch_size, max_sequence_length]. Equation 1-3 is the process of weighting operation.

In equation 1, $[Tf - Idf\_Output]_i$ is the result of the $i - th$ batch of Tf-Idf output. $[RoBERTa\_Output]_i$ is the result of the $i - th$ batch of RoBERTa output. The result of multiplying these two matrices is $[Weighted]_i$.

In equation 2, $[Tf - Idf\_Output]_i^T$ is the transpose of $[Tf - Idf\_Output]_i$ matrix. The result of multiplying $[Tf - Idf\_Output]_i^T$ and $[Weighted]_i$ is $[RoBERTa\_Weighted\_Output]_i$.

In equation 3, The value range of i is an integer between 0 and batch_size. Calculate the value of each $[RoBERTa\_Weighted\_Output]_i$ to get $[RoBERTa\_Weighted\_Output]$. Its shape is the same as RoBERTa output.

Figure 3 shows the model structure and data flow

Method	F1 Score
ALBERT+Tf-Idf	82.53
BERT+Tf-Idf	82.04
RoBERTa	83.41
RoBERTa+Tf-Idf	84.81

Table 2: F1 result scores obtained on the validation set using different models. The validation set is provided by the task organizer team.

of RoBERTa combined with Tf-Idf.

3 Experiment and Results

In this section, we will introduce the data preprocessing methods and experimental settings we used in the task and the final results.

3.1 Data Preprocessing

Combined with our analysis in the data description section, we remove the stop words of sentence pairs in the data. For the stop word list, we use the stopwords package provided by NLTK. To use the Tf-Idf algorithm to obtain the weighted output, and to ensure that the shape of the text encoding processed by the Tf-Idf algorithm is consistent with the output shape of RoBERTa, we have deleted the part of the text encoding that exceeds the maximum sentence length. For those less than the maximum sentence length for text encoding, we perform zero-padding operations. The encoding of Tf-Idf is obtained using the toolkit provided by gsim (Řehůřek and Sojka, 2010) [2].

In the data input, we use the [SEP] symbol to separate the sentence pairs together. Then use the [SEP] symbol to concatenate Lemma that appears in each sentence in the sentence pair. It should be noted that the three models we used in the experiment, BERT, ALBERT, and RoBERTa, are different in the division of symbols. Here, we use [CLS] and [SEP] uniformly for the convenience of description.

3.2 Experiment setting

As we introduced in the previous section, on the data set for this task, we use 4 different models to experiment with the result scores on the validation set. We adjust the parameters as much as possible to achieve the optimal results of each different model, so different models use different parameter combination settings.

Team	F1 Score	Rank
jaymundra	93.30	1
rohangpt	93.30	1
oyx	93.30	1
rohangpt	93.20	2
dipakam	92.80	3
LucasHub(our team 'hub')	84.60	49

Table 3: In the result list released by the task organizer team, the top 3 submitted test set prediction results scores and our submitted test set prediction results scores. There are a total of 175 results on the leaderboard of the English task. There are a total of 87 places from the first to the last.

- ALBERT+Tf-Idf: The epoch, batch size, maximum sequence length, and learning rate for the model are 6, 32, 150, and 3e-5, respectively.
- BERT+Tf-Idf: The epoch, batch size, maximum sequence length, and learning rate for the model are 4, 32, 150, and 4e-5, respectively.
- RoBERTa+Tf-Idf: The epoch, batch size, maximum sequence length, and learning rate for the model are 5, 32, 150, and 3e-5, respectively.
- RoBERTa: The epoch, batch size, maximum sequence length, and learning rate for the model are 5, 32, 150, and 3e-5, respectively.

4 Results

The final result score evaluation index uses the F1 score. Therefore, the effects of the different models we used in the experimental phase are all using F1 scores to determine which model is better.

We use the same validation set data to evaluate the performance of different models. Comparing the result score obtained by the combination of ALBERT, BERT and Tf-Idf with the score obtained by the combination of RoBERTa and Tf-Idf, it can be seen that the combination strategy of RoBERTa can get a better F1 score. Compared with the F1 score obtained by using RoBERTa alone, the F1 score obtained by RoBERTa+Tf-Idf is better. This also verifies the feasibility and effectiveness of our method. We sort the results according to Table 2.

The prediction result of the English test set we finally submitted is predicted by RoBERTa+Tf-Idf.

[2] https://github.com/RaRe-Technologies/gensim

Compared with the F1 scores obtained by the top three teams in the English data, there is still a certain gap. Our F1 score ranks middle among all result scores. Our final ranking is 49th. We sort the results according to Table 3.

5 Conclusion

This paper proposes a model that combines RoBERTa and Tf-Idf to calculate whether the target words in English sentence pairs are similar. We introduced our analysis of the data, the methods used in the experiment, and the results of the experiment in Sections 3 and 4. We compared the effects of different models of ALBERT, BERT, RoBERTa and the combination of Tf-Idf. The experimental results also prove that RoBERTa+Tf-Idf can get better results in our method. In future work, we will improve our methods to get better results. For example, other types of word embedding vectors can be introduced into our model, and the method of weighting and vector fusion can also be improved.

References

Oscar Araque, Ganggao Zhu, and Carlos A Iglesias. 2019. A semantic similarity-based perspective of affect lexicons for sentiment analysis. *Knowledge-Based Systems*, 165:346–359.

Weilong Chen, Xin Yuan, Sai Zhang, Jiehui Wu, Yanru Zhang, and Yan Wang. 2020. Ferryman at SemEval-2020 task 3: Bert with TFIDF-weighting for predicting the effect of context in word similarity. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 281–285, Barcelona (online). International Committee for Computational Linguistics.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations.

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*.

Federico Martelli, Najla Kalach, Gabriele Tola, and Roberto Navigli. 2021. SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation (MCL-WiC). In *Proceedings of the Fifteenth Workshop on Semantic Evaluation (SemEval-2021)*.

Sachin Mathur and Deendayal Dinakarpandian. 2012. Finding disease similarity based on implicit semantic similarity. *Journal of biomedical informatics*, 45(2):363–371.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

George A Miller and Walter G Charles. 1991. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

Michael Mohler and Rada Mihalcea. 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 567–575.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Juan Ramos et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. http://is.muni.cz/publication/884893/en.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*.

David Sánchez, Montserrat Batet, David Isern, and Aida Valls. 2012. Ontology-based semantic similarity: A new feature-based approach. *Expert systems with applications*, 39(9):7718–7728.

Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. 2015. Convolutional lstm network: A machine learning approach for precipitation nowcasting. *arXiv preprint arXiv:1506.04214*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.