Will_Go at SemEval-2020 Task 3: An Accurate Model for Predicting the (Graded) Effect of Context in Word Similarity based on BERT

Wei Bao[*]
Southeast University
willinseu@gmail.com

Hongshu Che[*]
Southeast University
hsche1222@gmail.com

Jiandong Zhang[*]
Southeast University
zjdx1998@gmail.com

Abstract

Natural Language Processing (NLP) has been widely used in the semantic analysis in recent years. Our paper mainly discusses a methodology to analyze the effect that context has on human perception of similar words, which is the third task of SemEval 2020. We apply several methods in calculating the distance between two embedding vector generated by Bidirectional Encoder Representation from Transformer (BERT). Our team *will_go* won the 1st place in Finnish language track of subtask1, the second place in English track of subtask1.

1 Introduction

Computing the meaning difference between words in the semantic level is a task that has been widely discussed. In the area of natural language processing (NLP) like information retrieval (IR), there are many specific applications using similarity, such as text memorization (Lin and Hovy, 2003), text categorization (Ko et al., 2004), Text Q&A (Mohler and Mihalcea, 2009), etc.

The task3 of SemEval-2020[1] focuses on the influence of context when humans perceive similar words (Armendariz et al., 2020a). As we all know, polysemous words have different meanings in a totally different context, which the current translation system can recognize very well. However, many translation systems can't exactly predict the subtle variance on the meanings of words, which is also caused by a different context.

Task3 has two sub-tasks. In subtask1, we are required to predict the extent of change in scores of similarity between two words in different contexts by human annotators. In subtask2, we only predict the absolute score as is in the subtask1 rather than the difference in scores, and we would only discuss subtask1.

Our team uses different algorithms to calculate the distance between two embedding vectors generated by BERT (Devlin et al., 2018) and defines it as the similarity. So we can get the change in similarity by subtraction between two distances. However, this methodology did not get exciting performance in the task evaluation, so we improve this by blending different BERT models, which we will introduce later in Section 3.2.

2 Related Work

There are many methods and models to estimate the similarity between long paragraphs. Most of them treat it as a binary classification problem, Hatzivassiloglou et al. (Hatzivassiloglou et al., 1999) compute the linguistic vector of features including primitive features and composite features, then they build criteria by feature vectors to classify paragraphs. As for similarity between short sentences, Foltz et al.(Foltz et al., 1998) suggest a method that provides a characterization of the extent of semantic similarity between two adjacent short sentences by comparing their high-dimensional semantic vectors, which is also a Latent Semantic Analysis (LSA) model. Both LSA and Hyperspace Analogues to Language

[*] These authors contributed equally to this work.

[1] https://competitions.codalab.org/competitions/20905

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 301–306
Barcelona, Spain (Online), December 12, 2020.

(HAL) (Burgess et al., 1998) are all corpus-based model, the latter one uses lexical co-occurrence to generate high-dimensional semantic vectors set, where words in this set can be represented as a vector or high-dimensional point so that their similarities can be measured by computing their distances.

Although computing similarity between words are less difficult than between texts, there still exist some sophisticated problems. Similarity between words is not only in morphology but more significantly in semantic meaning. The first step of reckoning the similarity between words is using Word2Vec(Mikolov et al., 2013a), which is a group of corpus-based models to generate word embedding, and mainly utilizes two architectures: continuous bag-of-words (CBOW) and continuous Skip-gram. In the CBOW model, the distributed representations of context are made as input to the model and predict the center words, while the Skip-gram model uses the center words as its input and predict the context, which predicts one word in many times to produce several context words. Therefore, the Skip-gram model can learn efficiently from context and performs better than the CBOW model, but it takes much more consumption in training time than the CBOW model. But since hierarchical softmax and negative sampling (Mikolov et al., 2013b) were proposed to optimize the Skip-gram model when training large-scale data.

Word2Vec cannot be used for computing similarity between polysemous words because it generates only one vector for a word, while Embedding from Language Model (ELMo) (Peters et al., 2018) inspired by semi-supervised sequence tagging (Peters et al., 2017) can handle this issue. ELMo is consist of bidirectional LSTM (Hochreiter and Schmidhuber, 1997), which makes the ELMo have an understanding of both next and previous word, it obtains contextualized word embedding by weight summation over the output of hidden layers. Compared with the LSTM used in ELMo, Bidirectional Encoder Representation from Transformer (BERT) (Devlin et al., 2018) is a stack of Transformer Encoder (Vaswani et al., 2017), which can be computed in parallel ways and save much time in training. There are two BERT versions with different size, one is BERT Base, which has 12 encode layers with 768 hidden units and 12 attention heads, and the other is BERT Large, which has 24 encode layers with 1024 hidden units and 16 attention heads, achieved state-of-the-art results according to that paper.

3 System Overview

3.1 Data

The source of our test data is from the CoSimLex dataset (Armendariz et al., 2019), which is based on the well known SimLex999 dataset (Hill et al., 2014) and provides pairs of words.

In task3, the English dataset consists of 340 pairs; the Finnish, Croatian, Slovenian consist of 24, 112, 111 pairs respectively. Here is the quantity count table.

Language	Abbr.	Count
English	En	340
Croatian	HR	112
Finnish	FI	24
Slovenian	SL	111

Table 1. Test Dataset in Task3 from CoSimLex dataset

Each language datafile has eight columns, namely *word1, word2, context1, context2, word1_context1, word2_context1, word1_context2, word2_context2*, and their meanings are first word, second word, first context, second context, the first word in the first context, the second word in the first context, the first word in the second context, the second word in the second context respectively. In addition, word1 and word2 may have a lexical difference between word1_context and word2_context.

3.2 Methodology

The BERT model architecture is based on a multilayer bidirectional Transformer as Figure 1.Instead of the traditional left-to-right language modeling objective, BERT is trained on two tasks: predicting randomly masked tokens and predicting whether two sentences follow each other. BERT model gets a lot of state-of-the-art performance in many tasks, and we also use the BERT model in our strategy. We approach this task as one of tasks that calculates the similarity between two words. In our model, context data would be firstly added into BERT like the following Figure 2.

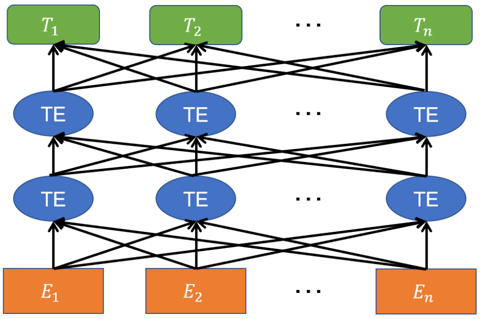

Figure 1. Bidirectional Transformer architectures of BERT

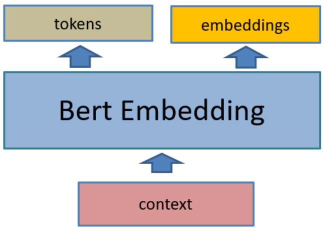

Figure 2. BERT schematic diagram

Inspired by CoSimLex(Armendariz et al., 2020b), our model would calculate the distance by several algorithms immediately when it obtained embedding of each token, then we predict the graded effect of context in word similarity as in the following steps:

- **Step1**: Choose the corresponding two embeddings of word1_context1 and word2_context1, compute the distance in several algorithms as SC_1.

- **Step2**: Substitute the words in Step 1 with word1_context2 and word2_context2, and repeat the last step, then we get the SC_2.

- **Step3**: By subtraction, we can get the change on similarity $C = SC_1 - SC_2$

- **Step4**: Change the distance computing algorithm and repeat Step1 $\sim$ Step3.

- **Step5**: After Step1 $\sim$ Step4, we can obtain a vector of change, $C_1, C_2, \cdots, C_n$, where n denotes the number of distance calculating algorithms used in our model and w_i denotes the manual parameter, we get the final change
$$C = \sum_i w_i C_i, \sum w_i = 1$$
.

Here we provide a flow chart Figure 3 to show the process from Step1 to Step4.

3.3 Experiment

We trained one standard BERT Large model and one multilingual BERT Base model by MXNet (Chen et al., 2015). The dataset we trained BERT Large model is *openwebtext_book_corpus_wiki_en_cased*, which were maintained by GluonNLP[2], and we trained Multilingual BERT (M-BERT)(Pires et al., 2019) Base model by *wiki_multilingual_uncased* dataset that also provided by GluonNLP. It takes much time to train the BERT model, so we recommend utilizing the well trained BERT model from bert-embedding[3].

After configuring the models, we can follow the section 3.2 by giving the input from section 3.1 and get the experiment results which will be introduced in Section 4. Task3 has four language tracks, namely English, Croatian, Finnish, Slovenian. We use the BERT Large model in the English track, and Multilingual BERT Base model in the other three tracks.

In section3.2, we use several algorithms to compute similarity. Here we introduce two main algorithms that used in our experiments.

[2] https://gluon-nlp.mxnet.io/model_zoo/bert/index.html

[3] We can simply use the model by pip or conda install bert-embedding

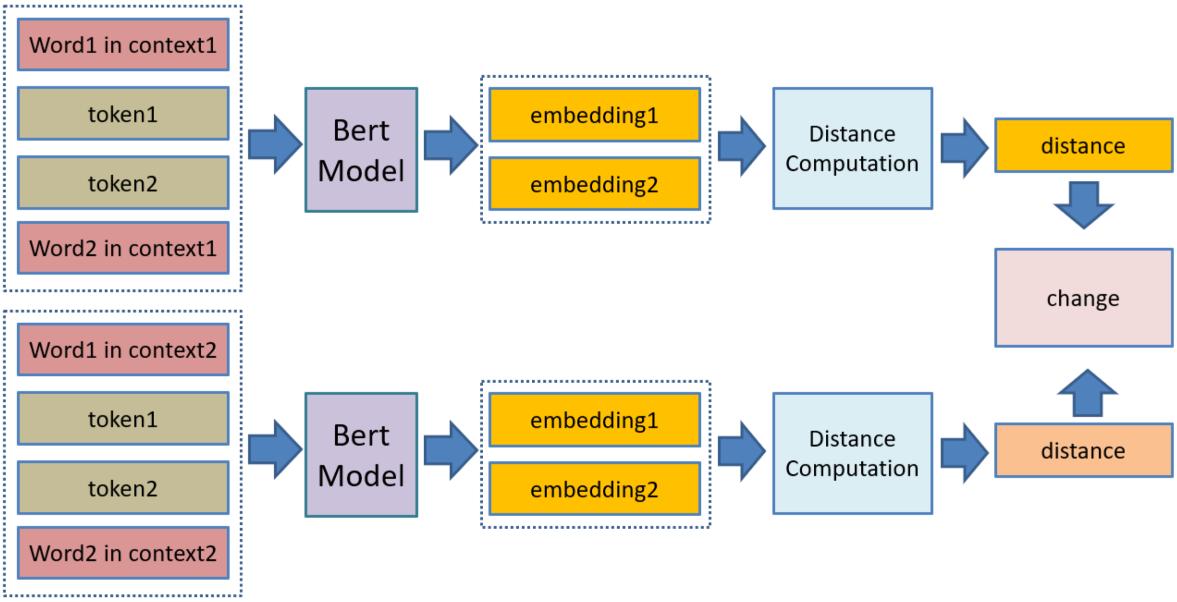

Figure 3. Part Flow Chart of our model

- Cosine Similarity that calculates the cosine of angle between two vectors.

$$sim(w_1, w_2) = \frac{w_1 \cdot w_2}{\|w_1\| \, \|w_2\|} = \frac{\sum_i w_{1i} w_{2i}}{\sqrt{\sum_i w_{1i}^2} \sqrt{\sum_i w_{2i}^2}}$$

- Euclidean Distance that calculates the square root of square distance in each dimension.

$$sim(w_1, w_2) = \sqrt{\sum_i (w_{1i} - w_{2i})^2}$$

4 Results

In our experiment targeted at subtask1, the English language track uses the Bert Large model, the Euclidean distance is 0.718 and the cosine distance is 0.752, the Blend result is 0.768, and the online LB ranks second; Croatian, Finnish, and Slovenian languages all use the Multi-lingual Bert model. The Croatian language track' Euclidean distance of 0.590, the cosine distance is 0.587, the Blend result is 0.594, and the online LB ranks sixth. The Finnish language uses the Euclidean distance of 0.750, the cosine distance is 0.671, the Blend result is 0.772, and the online LB ranks 1, The Slovenian language uses a Euclidean distance of 0.576, a cosine distance of 0.603, a Blend result of 0.583, and an online LB ranking seventh. We sort the result out the following Table 2.

Language & Abbr.	Model	Euclidean Dis.	Cosine	Blend	Rank
English, En	BERT Large	0.718	0.752	0.768	2
Croatian, HR	M-BERT Base	0.590	0.587	0.594	6
Finnish, FI	M-BERT Base	0.750	0.671	0.772	1
Slovenian, SL	M-BERT Base	0.576	0.603	0.583	7

Table 2. Experiment Results

5 Conclusion

In our paper, we propose a model that computes the similarity and similarity change by blending cosine similarity and euclidean distance, which calculated by two word embedding vectors. We firstly transform words in dataset that we introduce in section 3.1. into the word embedding vectors by BERT that we discuss in section 3.2, then we calculate the distance between two vectors, finally we blend the two distances computed by different algorithms as the final predict result. In the subtask1 of task3, our team *will_go* won a champion in Finnish track and the second place in English track.

References

Carlos Santos Armendariz, Matthew Purver, Matej Ulčar, Senja Pollak, Nikola Ljubešić, Marko Robnik-Šikonja, Mark Granroth-Wilding, and Kristiina Vaik. 2019. Cosimlex: A resource for evaluating graded word similarity in context.

Carlos S. Armendariz, Matthew Purver, Senja Pollak, Nikola Ljubešić, Matej Ulčar, Marko Robnik-Šikonja, Ivan Vulić, and Mohammad Taher Pilehvar. 2020a. SemEval-2020 task 3: Graded word similarity in context (GWSC). In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Carlos S. Armendariz, Matthew Purver, Matej Ulčar, Senja Pollak, Nikola Ljubešić, and Mark Granroth-Wilding. 2020b. CoSimLex: A resource for evaluating graded word similarity in context. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5878–5886, Marseille, France, May. European Language Resources Association.

Curt Burgess, Kay Livesay, and Kevin Lund. 1998. Explorations in context space: Words, sentences, discourse. *Discourse Processes*, 25(2-3):211–257.

Tianqi Chen, Mu Li, Yutian Li, Min Lin, Naiyan Wang, Minjie Wang, Tianjun Xiao, Bing Xu, Chiyuan Zhang, and Zheng Zhang. 2015. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Peter W Foltz, Walter Kintsch, and Thomas K Landauer. 1998. The measurement of textual coherence with latent semantic analysis. *Discourse processes*, 25(2-3):285–307.

Vasileios Hatzivassiloglou, Judith L Klavans, and Eleazar Eskin. 1999. Detecting text similarity over short passages: Exploring linguistic feature combinations via machine learning. In *1999 Joint SIGDAT conference on empirical methods in natural language processing and very large corpora*.

Felix Hill, Roi Reichart, and Anna Korhonen. 2014. Simlex-999: Evaluating semantic models with (genuine) similarity estimation.

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

Youngjoong Ko, Jinwoo Park, and Jungyun Seo. 2004. Improving text categorization using the importance of sentences. *Inf. Process. Manage.*, 40(1):65–79, January.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 150–157.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality.

Michael Mohler and Rada Mihalcea. 2009. Text-to-text semantic similarity for automatic short answer grading. pages 567–575, 01.

Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models.

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

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert?

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