BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2023
http://excel.fit.vutbr.cz

Answer Correctness Estimation on a Question

Marián Ligocký*

Abstract
A correctness estimation of an answer to a question is critical for a language learning applications where open answers are allowed. A possible approach is to compute semantic similarity of input sentence and predefined correct answers. Similarity score of two answers (sentences) can be computed by a deep learning models based on **transformer** architecture. We used models for finding **Semantic Textual Similarity** and **Natural Language Inference** between two sentences. The ability to derive meaningful score in a language-learning domain was examined. Bi-encoders and cross-encoders were compared. The best STS model improves correlation of scoring by 3.4% (Spearman) and 13.6% (Pearson). We advise to test our results with more data. Grammatical model might further improve scoring.

*xligoc04@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

[Motivation] Learning of languages was for a long time limited to closed questions (translating of words) with no voice answer possibility. Recent research allowed us to transcript speech to text. However scoring of sentence-alike answers is still difficult, since there are many correct answers to a question. A reliable model for scoring correctness of an answer to a question would improve possibilities of language teaching with no-human teacher.

[Problem definition] Let's have an set of predefined correct answers to a question. After student answers, his/her answer is compared with each of correct answers. Model determines whether the student's answer is correct and returns a score. Because no data were provided, to measure the correctness we decided to use Semantic Textual Similarity (STS) and Natural Language Inference (NLI) tasks.

[Existing solutions] Current state-of-art models for language processing are based on transformer architecture (BERT [1], RoBERTA [2], DeBERTA [3] and others). These models can be used for variety of NLP tasks, including STS (benchmark STSB [4]) and NLI (benchmarks MNLI [5] and SNLI [6]). Existing pretrained models are available on repositories (GitHub, Hugging Face).

[Our solution] We found out that it is not possible to use just one model for both STS and NLI. Better approach is to use two separate models. STS model we used improved correlation of model score to human-annotated score by 3.4% (Spearman) and 13.6% (Pearson). We recommend to enhance the system with an additional grammatical model.

[Contributions] Our work enhances the performance of current scoring system and sets recommendations for future development, including an additional grammatical model and better data that enables researchers to evaluate models reliably. Our work in the domain of language-learning application cannot be validated properly by today's data from authors of the app.

2. Poster description

2.1 Problem definition

Let's have an application for learning of languages, where you interact in a dialog, as shown in the first section. A prompt is given to you: "*Ask the neighbour to lower their music*". You answer by using your microphone and text transcript: "*Please, can you lower your music?*" is shown. Your answer is **compared literally** to predefined correct answers. If it is not among them, it is marked as **incorrect**. Our application uses a different method based on a neural network to **find similarity** of your answer to correct answers. If the similarity is high, then your answer is considered correct.

2.2 Word similarity via embeddings

How can a deep neural network find similarity among words? One approach is to create a vector representation of a word, **a word embedding**. This vector contains features of the word and vector arithmetic is possible, as shown in Figure 1. Popular tool **Word2Vec** [7] learns embeddings by predicting which words are likely to appear in the context of other words.

2.3 NLP tasks used to estimate a correctness to a question.

Since no data were given at the beginning of a work, we decided to make a model that solves a general tasks of Semantic Textual Similarity (STS) and Natural Language Inference (NLI). STS finds how semantically similar two texts are. Similarity is not defined exactly, even opposite sentences (contradictions) might have high degree of similarity. For detection of contradictions we use NLI, which labels whether given a first sentence (premise) the second sentence (hypothesis) are **entailment**, **contradiction** or **neutral**. The combination gives us two-dimensional score of similarity.

In the Table 1, a difference between a fast bi-encoder model *all-MiniLM-L6-v2*[1] and slower, but more precise cross-encoder *stsb-roberta-large*[2] is shown. Bi-encoder returns a high similarity score for a opposite sentence about increasing music (clear contradiction), whereas cross-encoder is better at the detection.

The Table 2 shows NLI labels for a predefined correct answer (premise) and student answers (hypothesis). All sentences are marked with correct labels. Cross-encoder is used with an classification head on top with three labels.

2.4 Model architecture. Bi-encoder vs. cross-encoder

Figure 2 shows a comparison of different architectures of BERT-like models. **Bi-encoder** is an architecture proposed in SBERT [8] which enables to compute embedding for each sentence individually. These embeddings are computed for each sentence only once and can be cached. It speeds up question answering and information retrieval. To find a most similar sentence pair from 1 000 sentences, model would run only 1 000x.

Cross-encoder would not be applicable for this kind of task, because it takes an input of two sentences and a classification head on top of BERT/RoBERTA/MPNet/... returns a score of similarity as probability of the label.

[1] Hugging Face repository: all-MiniLM-L6-v2
[2] Hugging Face repository: stsb-roberta-large

3. Results

Can one model do both tasks? No, we fine-tuned a bi-encoder model, previously pretrained on STS, to NLI task. Model was able to compute NLI labels, but did not perform well on STS.

Two models have to be used, one for STS and second for NLI. Cross-encoders have **better performance** in STS than bi-encoders, but does not produce embeddings and each pair of sentences has to be evaluated together. The best performing model *cross-encoder/stsb-roberta-large* improves previous scoring mechanism by 13.6% (Pearson correlation with respect to human-annotation). The difference in old Figure 3 and our scoring Figure 4 is evaluated on around 700 real student answers to around 100 questions with human-score evaluated by the creator of questions. The Figure 4 also includes NLI labels (green is entailment, black is neutral and red is contradiction).

4. Future work

Previously stated improvements have to be validated by an additional data generated by human annotators. Group of annotators might prevents bias. Grammatical model might help to improve score reliability. If enough data, cross-encoder models might be fine-tuned to match desirable score.

Acknowledgements

I would like to thank my supervisor Ing. Igor Szőke, Ph.D. for his help and the possibility to work on this interesting topic.

References

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

[2] 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. *CoRR*, abs/1907.11692, 2019.

[3] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced

BERT with disentangled attention. *CoRR*, abs/2006.03654, 2020.

[4] Daniel M. Cer, Mona T. Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1: Semantic textual similarity - multilingual and cross-lingual focused evaluation. *CoRR*, abs/1708.00055, 2017.

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

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

[7] Tomás Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013.

[8] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084, 2019.