COMP 490 - PROJECT REPORT

Axel Bogos - 40077502
Gina Cody School of Engineering and Computer Science
Concordia University
Montreal, Quebec, CA
a_ogos@live.concordia.ca

December 19, 2020

1 Introduction

During the Fall 2020 semester, I undertook a project under the supervision of Dr. Leila Kosseim and Mr. Reza Davari. This project consisted of a task from the SemEval 2021 workshop. SemEval is an international shared task on semantic evaluation organized by the Special Interest Group on the Lexicon of the Association for Computational Linguistics[1]. The task chosen is the task four of the 2021 workshop: *Reading Comprehension of Abstract Meaning*. This task consisted of bench-marking the reading comprehension capabilities of current models. The means by which this capability is measured is by predicting a missing word from a summary of a given text from a list of choices. That is, the input consists of a text, a summary where a particular word is removed and 5 options for the word that best represents the meaning of the text. A example is shown in table 1. The task is subdivided in three closely related sub-tasks: sub-task 1 requires the model to perform reading comprehension bench-marking as previously described where the missing word represent imperceptible concepts, such as *objective*, *culture* or *economy*. The example shown in table 1 belongs to this sub-task. sub-task 2 is concerned with reading comprehension where the missing word is a concrete object, like *groundhog*, or *chair*. Sub-task 3 evaluates the performance of a model trained on one of the sub-task and tested on the other, for instance a model trained to perform reading comprehension on imperceptible concepts and tested on text passages where the missing word is a concrete entity. Hence the difference in the sub-tasks lies mainly in the data used to train and evaluate the model.

Table 1: Input example

Passage	... observers have even named it after him, "Abenomics". It is based on three key pillars - the "three arrows" of monetary policy, fiscal stimulus and strctural reforms in order to ensure long-term sustainable growth in the world's third-largest economiy. In this week-end's upper house elections, ...
Question	Abenomics: The @**placeholder** and the risks
Answer	(0) chances (1) prospective (2) security **(3) objectives** (4) threats
Label	3

2 Data

The organizers have provided a trail data-set from the onset and a training data set on October 1st 2020. Each data-set contain specific files for sub-task 1 and 2. The trail data-set was to be used as a demonstration of the formatting of the input and to test the functioning of a model; while the training data-set is reserved to train the model. Although the test data-set was said to be made available on December 3rd by the organizers, it has not been made available. The

[1] https://semeval.github.io/SemEval2021/

organizers have not responded to a GitHub issue raised for the matter nor to a post on the competition forum. Hence all further discussion is based on using the trail data-set as a preliminary test-set. the trail data-set has an instance number ratio of 0.3 to the training data-set.

3 Baseline Model

A gated-attention reader [1] was provided by the organizers to be used as a baseline model for text comprehension. This model was to be used with a GloVe word embedding. Despite the best efforts of myself and Mr. Davari to run the baseline model, it could not be made to correctly run. An issue was opened on Github on November 28th, but has not yet been addressed or resolved by the organizers.

4 BERT

For this task, it was decided to use the BERT language model [2], which is short for Bidirectional Encoder Representations from Transformers. BERT models represent the input as 3 distinct embeddings: a tokens embedding, a segmentation embedding and a position embedding. The token embedding encapsulates the tokenized representation of words and special tokens that belong to the BERT model. BERT tokenization views words as made of sub-units which may be separated to better extract meaning or handle rare words; for instance, *walking* may be tokenized are *{walk, ##ing}*, where the double hashes represent that the token is a suffix. The BERT model also use special tokens such as *[CLS], [SEP]* and *[MASK]*. *[CLS]* is used to mark the beginning of an input, while *[SEP]* is used to mark the separation of an input in two sequences. In many applications including ours, there is a need to represent the relation of two sequences with each-other; for example in question-answering or, as in this task, reading comprehension. Finally, *[MASK]* marks a particular value for which we predict a replacement word; in our case *@placeholder*. The segmentation embedding represents to which sequence a token belongs, and finally the position embedding represents the order in which tokens appear in the input, hence adding word order information. BERT models are pre-trained on large unlabeled text ensemble and only require an additional output layer adapted to a particular task; this is referred to as the *fine-tuning phase*.

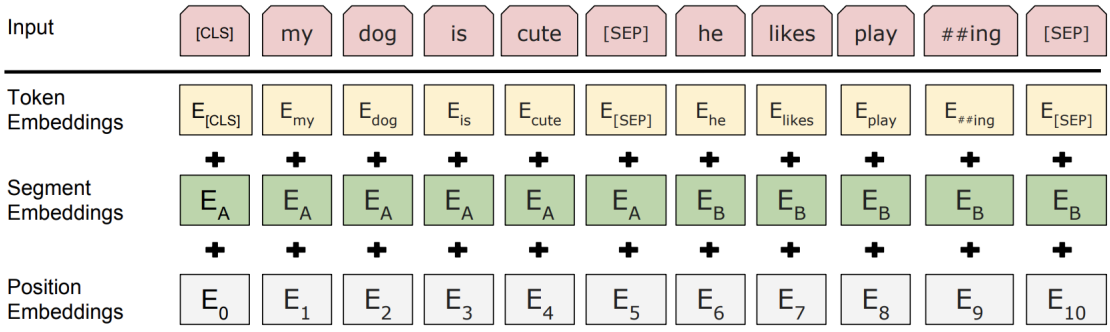

Figure 1: BERT input embeddings. Figure from Devlin et al. (2019) [2]

5 Data Preparation for a BERT model

In order to match the expected format of a BERT model, the input data was pre-processed. Notably, the "Passage" and "Question" columns were concatenated as a single column, with the aforementioned *[SEP]* token to discriminate between teh two sequences. BERT models accept a maximum of 512 tokens per instance which, counting the three special tokens, leave 509 tokens to be used for word representation. Hence during the pre-processing, inputs were temporarily tokenized, trimmed to a maximum of 509 tokens, and re-converted to strings. We finally replaced the *@placeholder* marker by a *[MASK]* token. Note that the *[CLS]* is automatically added at the beginning of the input during tokenization.

6 Modelling

The examples of the HuggingFace documentation served as a basis to the model; particularly the multiple-choice BERT model example[2]. Hyper-parameter search was done for multiple parameters; most notably the number of warm-up steps and the learning rate. Warm-up steps are a number of steps where the learning rate decrease according to a certain function, here in a linear manner, in order to converge more quickly and avoid local minima. After experimenting on Google Collaboratory GPUs, the project was moved onto a GPU equipped computer from the CLaC laboratory. An overview of the hyperparameters used in the best performing model is shown in table 2.

Table 2: Hyper-parameters of BERT model

Parameter	BERT
Learning rate	5E-6
Epochs	3
Max sequence length:	80
Batch size	1
Warm-up steps	800

7 Results

Due to the fact that the real test sets have not yet been made available, unfortunately all our results are preliminary. Also, due to the fact that the issues in the baseline code have not been addressed by the organizers, it is hard to assess the performances of the BERT model on their own. Nevertheless, here is what was obtained after hyper-parameter optimization. Note that accuracy was used as a measure of success as it is the sole criteria by which the competition judges submitted models according to the task description. The sub-task 3 is further sub-divided in sub-task 3.1 and sub-task 3.2, where the former is a model trained on data from sub-task 1 and tested on data from sub-task 2 and the latter is a model trained on data from sub-task 2 and tested on data from sub-task 1.

Table 3: Results

Sub-task	Accuracy of best model
1	0.491
2	0.4875
3.1	0.490
3.2	0.485

8 Analysis & Conclusion

Our results are unfortunately inconclusive: the lack of a baseline and the actual test sets prevents making definitive statements about the results obtained. However, we may still make observations from the results in relation to each-other. We first notice that there is not a significant difference in performance between sub-task 1 and 2, pointing to the fact that the BERT model is equally capable of addressing the reading comprehension of imperceptible and concrete entities regardless of whether or not our results are optimal in either task. Reinforcing this point is that sub-task 3.1 has similar results to sub-task 1 and sub-task 3.2 has similar results to sub-task2; hence it seems the differentiation between predicting word representing imperceptible or concrete entities is not significant. Nevertheless, this project was a significant learning opportunity. Progress was slower than initially anticipated as I faced a steeper learning curve than I imagined. Reviewing the literature, gaining insight in the BERT architecture and HuggingFace documentation but also gaining practical experience in working with remote GPUs for example have all been immense learning opportunities and motivating challenges. I believe I am much better equipped than four months ago to continue learning and work on further deep-learning projects.

[2] `https://huggingface.co/transformers/examples.html`

References

[1] Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. Gated-attention readers for text comprehension. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. *Proceedings of the 2019 Conference of the North*, 2019.