Comparing BERT Variants on NLP Tasks

A report on the performance of different BERT models

Author: Oscar Pang

Education: Holy Trinity School

Date: November 9, 2022

Contents

1 Abstract

In the last 5 years, we have witnessed significant enhancements in the field of of Natural Language Processing (NLP), beginning with the development of Transformers. Since then, many transformer-based machine learning techniques have been built, with the most prominent one beig the BERT model. After the invention of BERT, many computer scientists began to create variations of it, by modifying the open-source code. This report explores three popular variations (RoBERTa, DistilBERT, ALBERT) in detail and analyzes their performances when applied to sentiment analysis, extractive question answering, and classification for a dataset known as Word-in-Context. To perform the training and evluation steps in these experiments, the Pytorch framework in Python is used.

2 Introduction

2.1 Natural Language Processing

Natural language processing (NLP) is the branch of machine learning concerned with giving computers the ability to understand text and spoken words in the same way that humans do. NLP combines computational linguistics with statistical, machine learning, and deep learning models [1]. When these technologies are combined, computers can process human language in the form of text or voice data and comprehend not only its meaning, but also the intent and sentiment of the speaker or writer. NLP has become part of our everyday lives, with virtual assistants like Alexa and Siri, Google search prediction, text autocorrect, and customer service chatbots. However, along with these applications come challenges. A primmary issue is with the context of words. The same words and phrases can often take on different meanings depending on the context of a sentence and many words. Literary techiques like iroy and sarcasmm further confuses NLP models, as they could sigificantly change the meaning of phrases. Another problem is the inability to incoporate langugaes that are less common, due to the lack of resources. New multilingual models have been developed to leverage universal similarities that exist between languages [2].

2.2 Transformers

In the paper "Attention Is All You Need," a novel architecture called Transformer is introduced. It makes use of attention mechanisms, as the title suggests. Transformer is an architecture for transforming one sequence into another using two parts: the encoder and decoder. However, it differs from previously existing sequence-to-sequence models in that it does not use Recurrent Neural Networks. Instead, Transformers use non-sequential processing: sentences are processed as a whole rather than word by word [3]. Transformer models have all been taught to be language models, indicating that they received extensive unsupervised training using a large volume of raw material.

Table 1: Characteristics of BERT Variations

Model	Parameters	Layers	Methods
BERT (base)	110M	12	Bidirectional Transformer, MLM, NSP
RoBERTa (base)	125M	12	BERT without NSP, Dynamic Masking
DistilBERT	66M	6	BERT with distillation
ALBERT (base)	12M	12	BERT with less parameters, SOP instead of NSP

2.2.1 BERT

Bidirectional Encoder Representations from Transformers, or BERT, is a state-of-the-art deep learning model that is based on Transformers and pre-trained using Wikipedia. It was developed by Google in 2018 and has revolutionized search engines, in terms of predicting user intentions

and indexing contents. In the past, language models, such as Long short-term memory (LSTM), could only interpret text input sequentially but not simultaneously. BERT is unique since it can simultaneously read in both directions. It is pre-trained on two different but related NLP tasks: Masked Language Modeling (MLM) and Next Sentence Prediction using this bidirectional capacity (NSP). The goal of MLM training is to conceal a word in a sentence, after which the programme will infer the hidden word from the surrounding context. The goal of NSP is to have the software determine if two provided sentences connect logically and sequentially or whether their relationship is just arbitrary [4].

2.2.2 RoBERTa

RoBERTa (Robustly Optimized BERT Pretraining Approach) is a BERT variant developed to improve the training phase. RoBERTa was created by training the BERT model on larger datasets of longer sequences and large mini-batches. RoBERTa researchers significantly improved their results by modifying BERT hyperparameters. BERT uses static masking, meaning that the same part of the sentence is masked in each epoch. In contrast, RoBERTa uses dynamic masking, wherein for different epochs, different part of the sentences are masked. This helps to improve the robustness of the model. RoBERTa also does not use NSP [5].

2.2.3 DistilBERT

DistilBERT is a distilled BERT-based Transformer model that is compact, quick, and computationally inexpensive. The goal is to reduce the size of BERT while still maintaining most of the accuracy. In order to shrink a BERT model during the pre-training stage, knowledge distillation is used. To leverage the inductive biases learned by larger models during pre-training, the authors of this model also introduced a triple loss combining language modeling, distillation and cosine-distance losses [5].

2.2.4 ALBERT

By utilizing parameter sharing and factorizing approaches, ALBERT (A lite version of BERT) was recently introduced to improve the training and outcomes of the BERT architecture. The BERT-based version has roughly 110 million parameters, which makes it difficult to train and affects computation when there are too many parameters. ALBERT, which has much fewer parameters than BERT, was developed to address this issue. Like RoBERTa, ALBERT also does not include NS. Instead, it utilizes Sentence Order Prediction (SOP) [5].

2.3 NSP vs SOP

The primary distinction between NSP and SOP is that during training, the model for NSP receives input pairs of sentences and learns to predict whether the second sentence will come after the previous one in the original text. The purpose of the SOP is to ”classify” if the two given sentences have been switched or not, or whether they are in the correct order [6].

3 Sentiment Analysis

3.1 Description of Task

Sentiment analysis, a type of text classification, is one of the most popular NLP tasks. It is the process of categorizing data into either binary labels based on its sentiment: positive (1) or negative (0), or rating labels: 1 star to any number of stars. Companies can apply sentiment analysis to a wide range of applications, including social media analysis, survey feedback, and product reviews. For this task, I used the IMDB movie reviews dataset with binary labels from the Hugging Face library.

3.2 Processing and Fine-tuning

For this task, I first separated the IMDB dataset into training and validation data, with 25000 examples in each. I processed this dataset by first tokenizing both the training and validation datasets using the AutoTokenizer from Transformers. I repeated this process for 4 different tokenizers: BERT, DistilBERT, ALBERT, and RoBERTa. While tokenizer, I also truncated the datasets so that the max length for each review was 50 tokens. Regarding the fine-tuning aspect, I set the batch size to 16 and the learning rate to 2e-5, which were the default values used in the Hugging Face sentiment analysis guide. Additionally, I set the number of epochs as 5, computing the training and validation loss, as well as the accuracy on the validation dataset for every epoch. The final accuracies after all 5 epochs were considered as the results.

3.3 Results

Table 2: Sentiment Analysis Performance

Model	Training Time	Final Accuracy
BERT	32:27	0.81704
RoBERTa	31:06	0.83960
DistilBERT	15:22	0.80108
ALBERT	29:06	0.80676

As shown in the table, RoBERTa performs the best in terms of final accuracy, while also taking less time compared to BERT. DistilBERT was the fastest to train, using less than 50 percent of BERT's training time but still maintaining around 98 percent of its accuracy. This means that, if we factoring in both training time and final accuracy, DistilBERT easily wins. As well as that, ALBERT beats BERT, as although its final accuracy is lower by around 1 percent, it is around 20 percent faster.

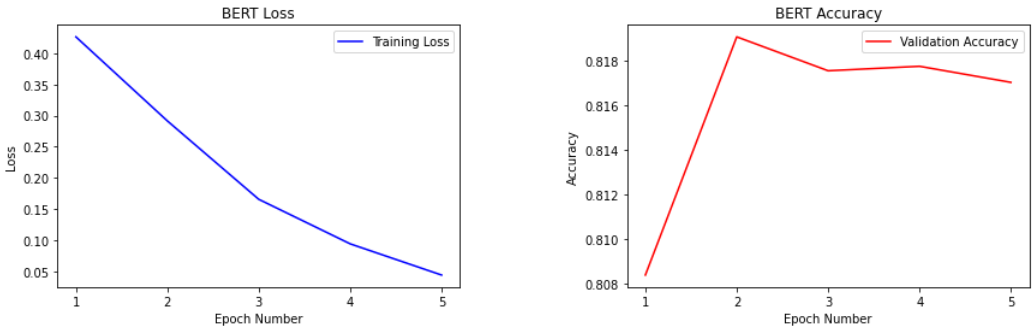

Figure 1: BERT Model Performance

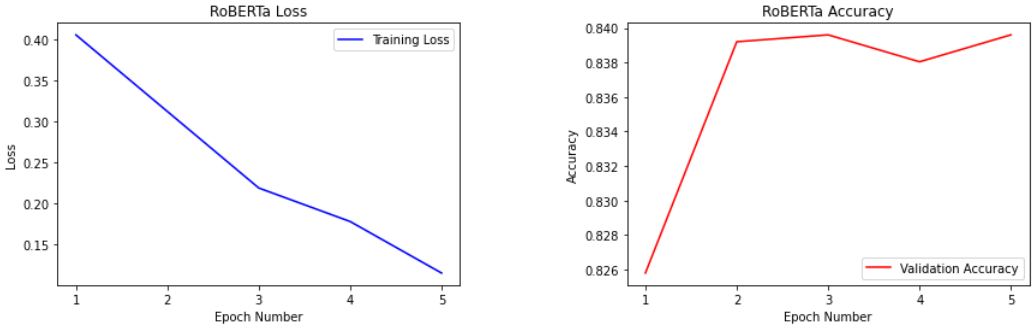

Figure 2: RoBERTa Model Performance

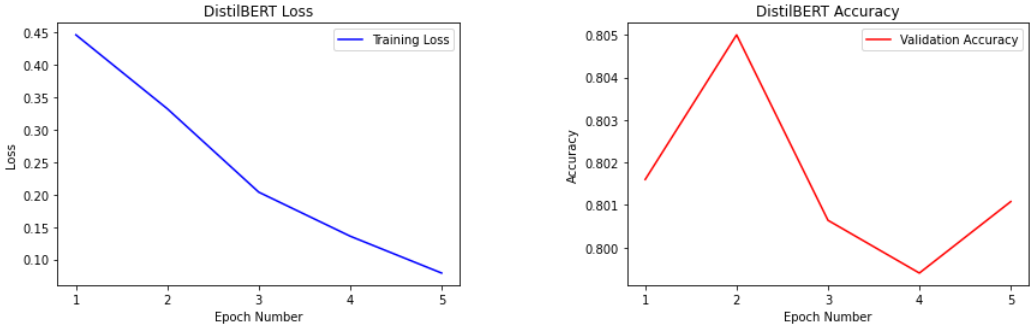

Figure 3: DistilBERT Model Performance

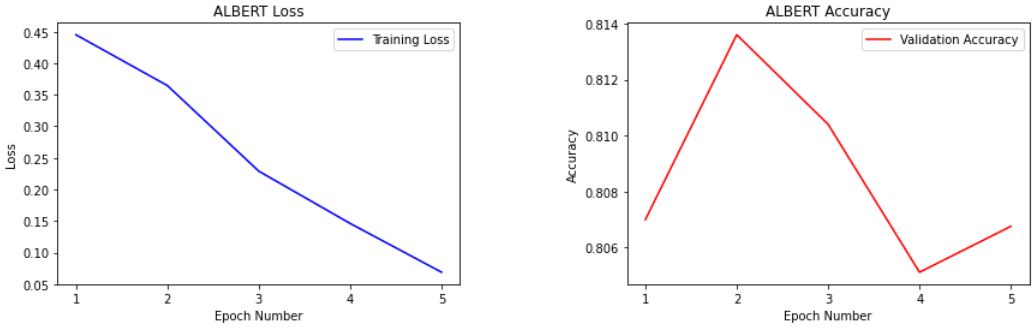

Figure 4: ALBERT Model Performance

4 Extractive Question Answering

4.1 Description of Task

Question answering (QA) is a branch of artificial intelligence within the natural language processing and information retrieval fields that involves the development of systems that answer questions posed by humans in natural language. Question answering programmes can generate answers by querying a knowledge base (a structured database of knowledge) or an unstructured collection of natural language contexts. For this task, I used the Stanford Question Answering Dataset (SQuAD) from Hugging Face, which is a dataset that focuses on questions that can be answered directly from a context (extractive).

4.2 Processing and Fine-tuning

Compared to the sentiment analysis task, there are more factors to consider when pre-processing a dataset for question answering. In order to reduce training time, I took 25 percent of data from both the default training dataset and validation dataset in the SQuAD, reducing the training dataset to 22742 examples and the validation dataset to 2741 examples. Additionally, I truncated each context to a maximum length of each context to 200 tokens, further reducing the time consumption. Something to note is that a standard evaluation metric would not work for a question answering model, as the outputs are not just single labels. Instead, the outputs consist of start logits and end logits for the answers to the questions. Therefore, we have to use one of these 2 metrics: Exact Match (EM), which evaluates how many times the model's predictions exactly match the characters of the true answers, and F1 Score, which is computed over the individual words in each prediction against those in the true answers [7]. For calculating the accuracy of each model, we will take into account both of these computing metrics.

4.3 Results

Table 3: Extractive Question Answering Performance

Model	Training Time	Final Accuracy (EM)	Final Accuracy (F1)
BERT	39.51	0.72256	0.81075
RoBERTa	42.22	0.82135	0.89508
DistilBERT	21.02	0.68471	0.77925
ALBERT	37.33	0.77441	0.85505

The first thing to note is that a higher Exact Match accuracy correlates to a higher F1 accuracy, and vice versa. Examining the results, this time, RoBERTa significantly surpasses all the other models in terms of final accuracy. However, it did also utilize the most time during the training phase. DistilBERT once again wins when it comes to training speed. However, unlike the sentiment analysis task, ALBERT performed better than BERT this time.

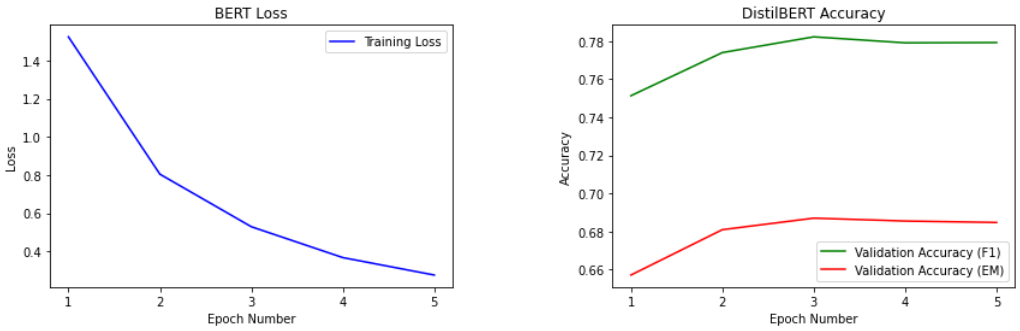

Figure 5: BERT Model Performance

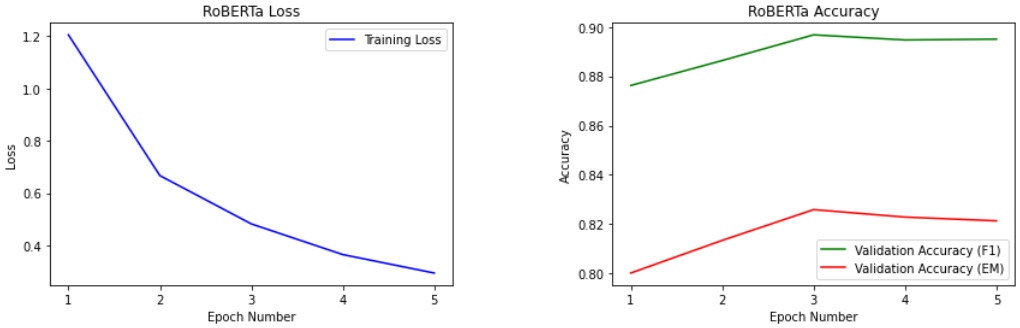

Figure 6: RoBERTa Model Performance

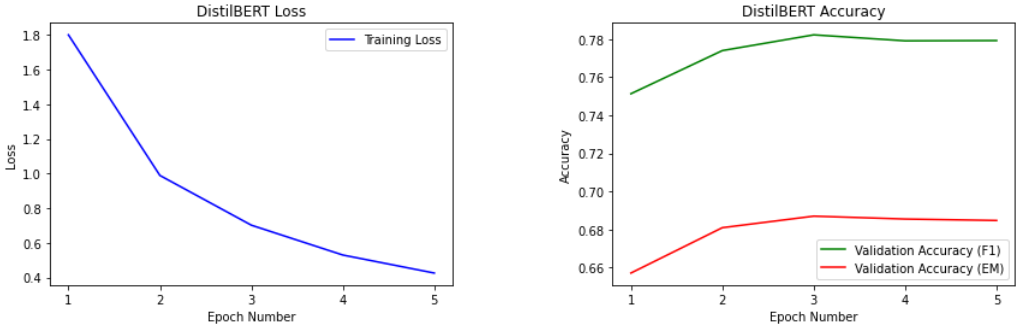

Figure 7: DistilBERT Model Performance

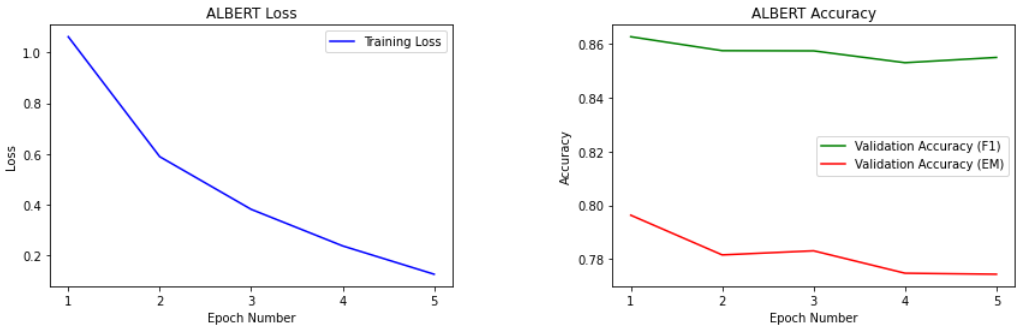

Figure 8: ALBERT Model Performance

5 Word-in-Context Classification

5.1 Description of Task

In this last part of the experiment, we look at a special task with a smaller custom dataset — the Word-in-Context Dataset (WiC). An ambiguous term can have several unrelated meanings depending on its context. The WiC dataset's objective is to determine each word's intended meaning. This is, similar to the sentiment analysis task, a binary classification task. Every instance of WiC contains a target word, which can be a verb or a noun and for which there are two contexts given. In each of these situations, the target word takes on a particular meaning. The goal is to identify if the occurrences of w in the two contexts correspond to the same meaning [8].

5.2 Processing and Fine-tuning

In comparison to the datasets for the previous tasks, this dataset is quite small, with only 5428 rows oxf training data and 638 rows of validation data. Each row contains a label (either true or false), a target word, the tense of the target word, the indexes of the target word in the first context sentence and second context sentence, and the 2 context sentences for the target word. To pre-process the dataset, I first converted the "true or false labels" to 0's and 1's; then I tokenized the sentences and padded them to a max length of 50 words, as most of the sentences do not exceed 50 words. To fine-tune the model, I created a class called WiCModel, with a forward propagation function that passes in self, the labels, the input ids, the attention mask, index1 (target word index of sentence 1), and index2 (target word index of sentence 2), as the parameters. For this task, I reduced the batch size to 8 and the number of epochs to 10 to get the best results, due to the small train data size. The learning rate was kept at 2e-5.

5.3 Results

Table 4: Word-in-Context Classification Performance

Model	Training Time	Final Accuracy
BERT	4:13	0.63950
RoBERTa	3:57	0.59718
DistilBERT	2:26	0.57994
ALBERT	4:50	0.61442

The results for this task differs from the other 2 tasks. RoBERTa no longer performs the best, and instead, BERT does. This could be due to the fact that RoBERTa does not use neither NSP nor SOP, while BERT and ALBERT do, respectively. As well as that, this time, ALBERT's training time exceeded that of BERT's by over 14 percent. This is conterintuitive, as ALBERT is supposed to be a "light" version that trains much quicker. Finally, as expected, DistilBERT is, once again, the fastest to train.

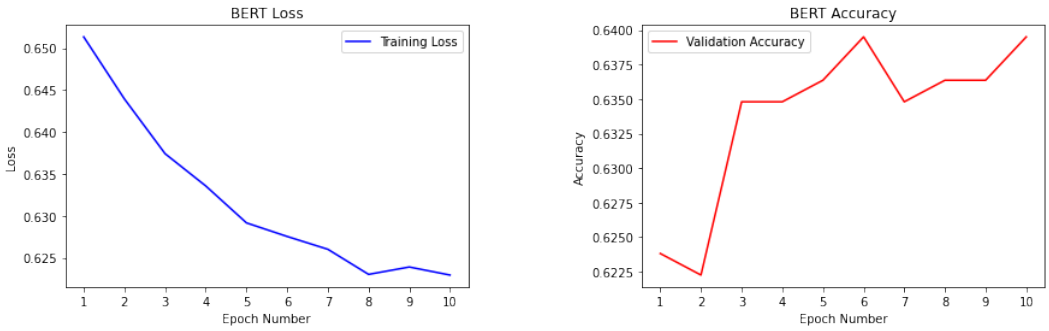

Figure 9: BERT Model Performance

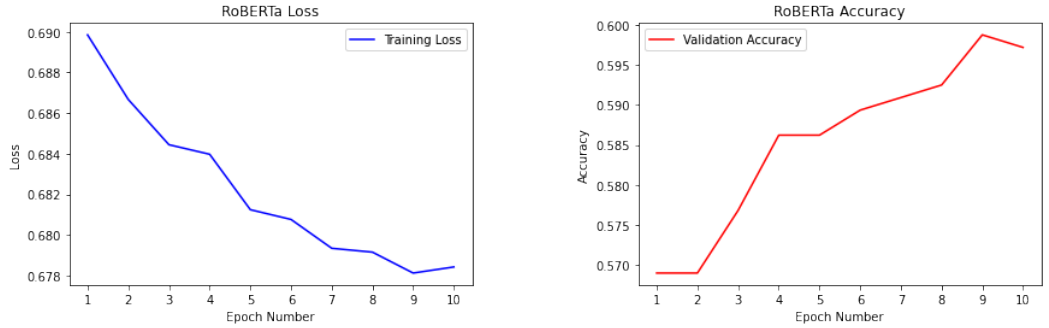

Figure 10: RoBERTa Model Performance

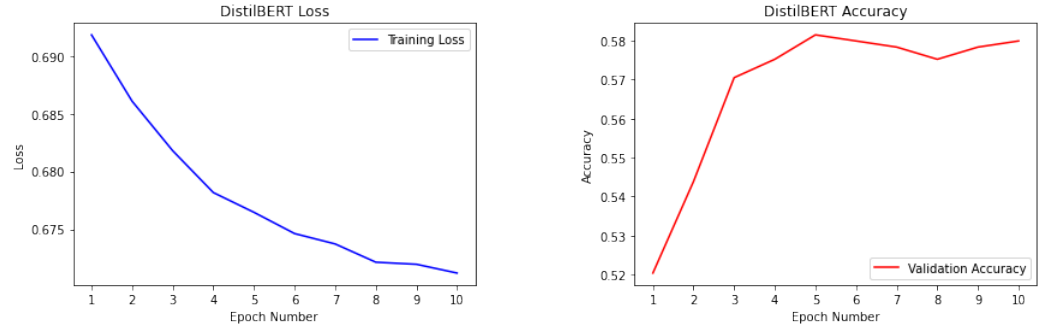

Figure 11: DistilBERT Model Performance

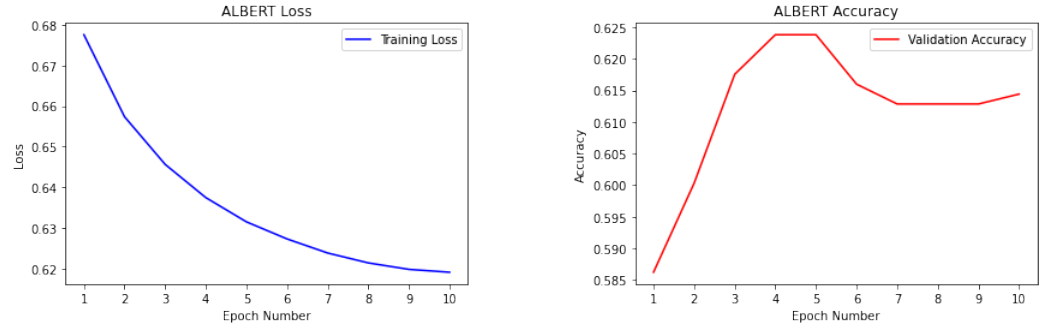

Figure 12: ALBERT Model Performance

6 Conclusion

Examining the results, we see that there is no single winner. While RoBERTa performs the best in Sentiment Analysis and Question Answering, it falls short behind both BERT and ALBERT in the Word in Context task. While DistilBERT may be the fastest model to train in all 3 tasks, it is also the least efficient all 3 tasks. It all depends on the task at hand and whether the goal is accuracy or speed. Another conclusion that can be drawn is that there are indeed noticeable differences in the training time and accuracy for each model for each situation, which indicates the importance of selecting the right model to use based on the elements of the task at hand. As we advance into the future, more and more BERT variations and new NLP models will emerge, improving upon certain aspects of the existing ones. Every new innovation in this field is a step closer to unlocking the maximum potential of artificial intelligence in linguistics.

References

[1] I. C. Education, "Natural language processing (nlp)," July 2, 2020. [Online]. Available: https://www.ibm.com/cloud/learn/natural-language-processing

[2] I. Roldós, "Major challenges of natural language processing (nlp)," December 22, 2020. [Online]. Available: https://monkeylearn.com/blog/natural-language-processing-challenges/

[3] Maxime, "What is a transformer?" January 4, 2019. [Online]. Available: https://medium.com/inside-machine-learning/what-is-a-transformer-d07dd1fbec04

[4] B. Lutkevich, "Bert language model," January, 2020. [Online]. Available: https://www.techtarget.com/searchenterpriseai/definition/BERT-language-model

[5] 360DigiTMG, "Bert variants and their differences," July 23, 2021. [Online]. Available: https://360digitmg.com/bert-variants-and-their-differences

[6] C. Durgia, "Exploring bert variants (part 1): Albert, roberta, electra," December 21, 2021. [Online]. Available: https://towardsdatascience.com/exploring-bert-variants-albert-roberta-electra-642dfe51bc23

[7] M. Beck, "Evaluating qa: Metrics, predictions, and the null response," June 9, 2020. [Online]. Available: https://github.com/fastforwardlabs/ff14_blog/blob/master/_notebooks/2020-06-09-Evaluating_BERT_on_SQuAD.ipynb

[8] M. T. Pilehvar, "Question answering," Apil 27, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1808.09121