CS 6241 Final Report

George Karagiannis (gk446) Himank Yadav (hy539)

May 21, 2020

1 Introduction

Unfortunately, there is an exciting and prevalent problem that our society currently faces; misinformation about COVID-19. Given the fact that the depth of our general knowledge about the pandemic is so limited and the rapid progression of affairs, the amount of false claims concerning the new pandemic is a real concern. We propose a novel approach in order to mitigate this problem, by applying automated Fact Checking to claims concerning the virus. In order to tackle this difficult and widespread problem, we apply numerical methods and language models. Early results indicate that we can achieve high precision through our anti-knowledge base of COVID-19 facts, however, we cannot make strong guarantees on recall due to the nature of this problem. Overall, we show whether we can use Language Models fine tuned on generic Natural Language Inference tasks to Fact Check claims about the Coronavirus.

We have released a Web interface of the proposed system, which can be found here: `https://akb-demo-271613.appspot.com/`. Note that the web interface uses a slightly older version of the AKB with less entries than the one presented in this report.

2 Problem Description

The scope of this problem revolves around data collection, analysis and prediction. The goal of this project is to conduct automated Fact Checking on unseen generic claims related to COVID-19 posed in natural language. Thus, we expect to classify a new claim as “false” when it is factually incorrect and also provide a justification for our prediction.

First, we created a new dataset that contains information about the virus by combining data from the Google Fact Checking API and the Poynter COVID-19 Database. Both of these sources contain human-labeled data from claims posed on social media and the Web in general. This real world dataset consists of roughly 4100 **false** claims about COVID-19.

Second, we apply a probabilistic model, which approximates the probability that a new claim is incorrect. One of the difficulties we face is that claims which need to be checked are posed in natural language and are inherently unstructured and hard to parse. For example, differentiating between claims with few differences structurally and big differences semantically has always been very challenging for NLP researchers. The claims “There is a vaccine for COVID-19” and “There isn’t a vaccine for COVID-19”, are hard to tell apart, but necessary for solving such problems.

3 Related Work

We plan on using BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. BERT was proposed by Devlin et. al. in 2018 and gained remarkable prominence due to its results on most NLP tasks, beating existing state-of-the-art methods. Under the hood, BERT uses the Transformer model and learns bidirectional-contextual relationship between words in a corpus of text. The original paper introduced two novel pre-training objectives: masked language model and next sentence prediction. In order to train the Masked Language Model, BERT’s training hides 15% of the input word tokens in a randomized order and then trains by trying to predict these masked tokens. These latent vectors which represent the masked tokens go into an output softmax layer over the entire training vocabulary. A big difference as opposed to traditional language models is that BERT does not estimate the conditional probability of the masked word given a directional context. Moreover, BERT also pre-trains on the next sentence prediction task, which allows it to understand temporal relationships between two given sentences. The original BERT model is pre-trained on book corpus (800 million words) [9] and English Wikipedia corpus (2500 million words). A big difference from prior state-of-the-art approaches, such as GPT and ELMo [7] is that BERT is pre-trained on bidirectional representations with a joint left and right conditioning across all layers. Even though the underlying architecture is common across all three approaches and is based on the transformer, BERT is able to significantly outperform the existing state-of-the-art approaches.

A BERT-like approach called RoBERTa proposed by Liu et al [5] achieves SOTA performance on most downstream classification tasks. RoBERRTa follows the same architecture as BERT with the exception of the NSP pre-training task. The main reason RoBERTa achieves a better performance lies in the larger pre-training corpus used, which comprises of 160 GB of text, contrary to 16 GB of text used in BERT.

The Multi-Genre Natural Language Inference (MultiNLI) corpus [8] is a labeled sentence pair dataset proposed by Williams et al. in 2018 for understanding textual entailment and contradiction. The corpus is modeled on the SNLI corpus [1], but differs in that it covers a range of genres of spoken and written text and supports a distinctive cross-genre generalization evaluation. The MNLI corpus contains 433 thousand pairs of sentences and is almost two orders of magnitude larger than all similar corpuses. Experiments with different models trained on NLI indicate that MNLI is a harder dataset than SNLI, as it includes ten distinct genres of written and spoken English, making it possible to evaluate systems on nearly the full complexity of the language. The authors baseline a variety of natural language inference models, including Continuous Bag of Words (CBOW), (Bi)LSTM and Enhanced Sequential Inference Model (ESIM) models. The authors report that at the time of writing, ESIM achieves the best performance among the neural network models and also does well on downstream natural language inference tasks. However BERT-based models like RoBERTa [5] now achieve SOTA performance on MNLI.

Additionally, previous work on "Anti-Knowledge Base" (AKB) [4] proposed by Karagiannis et al. introduces the use of a Database containing **false** claims, which can be used for Automated Fact Checking. The reasons of using an AKB are twofold. First, there do not exist any Knowledge Bases, which contain information about COVID-19 and most work by professional fact checkers is focused on finding mis-information concerning the virus. Second, a match (entailment) between an AKB entry and a test claim, will guarantee the untruthfulness of that claim, whereas the absence of a match does not guarantee truthfulness.

4 Approach

Consider the following setup. We have constructed an AKB, which consists of a set of n entries $\{e_i\}_{i=1}^{n}$ of sentences in natural language containing factual mistakes. For a given test claim c, the goal it to check if c is a factual mistake. We denote as $e_j \rightarrow c$ if $\exists\ 1 \leq j \leq n$ such that e_j entails c and as $e_j \not\rightarrow c$ otherwise.

1. We make use of existing research on Fact Checking, where we construct an "Anti-Knowledge Base" (AKB) [4], which contains **false** claims about COVID-19. Since the AKB contains factual mistakes, if a new sentence (which we want to fact check) can be logically derived from **any** of the claims in the AKB, then this sentence also constitutes a factual mistake. The fact that the majority of the hand-annotated COVID-19 related data contain false claims, motivated us to create an AKB. It is important to note that the goal of our approach is to detect factual mistakes but make no attempt at classifying a claim as correct. This means that for $1 \leq j \leq n$ if $e_j \rightarrow c$, then we classify c as a mistake, but if $e_j \not\rightarrow c$, we do not classify c as neither a mistake nor a correct statement. This is justified by the scope of our project, which revolves around finding mistakes about COVID-19.

2. We model the prediction of factual mistakes as a Natural Language Inference (NLI) task, where the goal is to determine whether a "hypothesis" is true (entailment), false (contradiction), or undetermined (neutral) given a "premise". In our use case, the AKB will contain a set of premises and a new claim (which we want to fact check) will be a hypothesis. If the hypothesis is **entailed** by **any** of the premises, then this hypothesis will be classified as a factual mistake.

3. We enhance the capability of the pre-trained BERT model by further fine-tuning it on the MNLI corpus. We will use this fine-tuned model to predict

Table 1: Examples of Entries obtained from GFC API and Poynter

Entry	**Rating**
Coronavirus was infecting people in the US since November 2019	FALSE
The coronavirus was created and patented by an American lab two years before the pandemic	FALSE
U.S. President Donald Trump implied the Obama administration left behind ”bad,” ”broken,” and ”obsolete” COVID-19 diagnostic tests	TRUE
A charitable hospital in Pakistan charged patients for novel coronavirus tests	MISLEADING

textual entailment for a new COVID-19 related claim.

4. To evaluate our method, we construct a test set for evaluation from two primary real-world sources. Our AKB contains **false** claims, but we also need **true** ones. To do that, we use again the Google's Fact Checking API and the the Poynter COVID-19 Database to retrieve the few **truthful** sentences about COVID-19 that we have not incorporated in our training process. Our evaluation criteria is binary to test the performance of our fact-checking model, which validates whether a given test statement is factually incorrect. Due to the nature of the problem, our primary goal is to achieve a high precision. However, we expect our system to have low recall, because of limited dataset. Precision accurately represents the quality of our fact-checking approach since we'll be able to accurately detect false sentences while calculating recall depends on the vast number of topics the sentences can include, which is beyond the scope of this project.

4.1 Dataset Description

Using as main sources the Google Fact Checking API and the Poynter Database both queried on COVID-19, we were able to obtain a total of 5320 claims posed in Natural Language along with the “rating” from the professional human fact checkers. Unfortunately, there does not exist a single schema for these ratings, which usually vary from “TRUE”, “PARTLY TRUE”, “FALSE””, “PARTLY FALSE”, “MISLEADING” and variations of those. Table 1 shows some examples of claims that are obtained from the two sources mentioned above. To create our AKB, we only keep claims with “FALSE” rating. Overall our AKB consists of 4139 claims, with an average length of approximately 16 words per claim. We are also interested in claims with “TRUE” rating, which will constitute part of our test set, when we evaluate our model. We were able to extract 32 truthful claims, with an average length of 19 words per claim. As stated above the number of false claims about COVID-19 is much greater than the number of true claims, which justifies our choice of using the AKB approach described above.

As we can see, the main limitation of the dataset is its size. We believe that 4139 AKB entries is probably not enough to cover the whole spectrum of all possible COVID-19 related claims. Hence, the recall of the system is expected to be low, since no premises will exist in the AKB to entail some of the test claims. Because of this lack of coverage, precision of our system should be the main metric of evaluation of our system. Despite its small size, we think its a reasonably-sized dataset to conduct experiments, such that we can evaluate the feasibility of our approach in terms of precision.

Furthermore, we plan on creating a test set which consists of the 32 truthful claims described above and 50 randomly selected factually incorrect claims randomly selected from the AKB. In order to have a balance of labels (“TRUE” and “FALSE”) we will also manually add 18 truthful claims derived from trustworthy sources. As a result, our test set will be comprised of total of 100 claims with 50 truthful and 50 factually incorrect.

4.2 Data Analysis

The Data Analysis part of the project concerns fine turning RoBERTa on the MNLI dataset and being able to predict textual entailment by using our AKB as a set of premises and the test claims as hypotheses. To fine tune on NLI, we fit a feed-forward Neural Network trained on MNLI, which contains approximately 500k premise-hypothesis pairs. The trained model, is used to predict textual entailment, as described above.

Despite fine-tuning, we use different word/sentence embedding approaches in order to find the top k most similar AKB entries for given a hypothesis. We experiment with different alternatives like Glove [6], BERT [3], Universal Sentence Encoder (USE) [2] and ELMo [7].

4.3 Numerical Method

As discussed in the previous section, we fine-tune the BERT model and use word embedding for sentences. However, our next big challenge was to accurately fact-check in a reasonable amount of time. A naive approach of running inference on every pair of sentences is infeasible given the size of our corpus. To get around this, we compute a multi-dimensional embedding representation of our corpus and computing cosine similarities of input sentences against our corpus.

$$\cos\theta = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\|\vec{b}\|} = \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2}\sqrt{\sum_1^n b_i^2}}$$

where

$$\vec{a} \cdot \vec{b} = \sum_{1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

is the dot product of the two vectors.

We then select the top k embeddings on which we run the downstream inference task. The word embeddings capture the semantic meaning of a given claim and taking the cosine similarity between such claims provides us with a set of most similar claims. Amongst this set, by selecting the top k, we intuitively select the top k most-related claims.

It is important to have a reliable similarity metric, because not all AKB entries are relevant to a test claim and it is computationally wasteful to predict textual entailment for all AKB entries, test claim pairs. Our preliminary studies suggest that this approach works well, since the pair of sentences used in comparison have their distances bounded in the embedding space. This allows us to offer fact-checking prediction in a feasible manner.

The main difference between Glove word embeddings and BERT and ELMo is that Glove embeddings are context independent and do not account for the context of the word in a sentence. The embedding vector produced by Glove embeddings discounts contexts such as the location of the word in a given sentence and its various potential meanings. A big potential downside here is that the same word can be used in multiple contexts in a given sentence, but Glove would not be able to capture the appropriate meaning which would be highly dependent on the context and word location. ELMo and BERT on the other hand generate context-dependent word embeddings, and can therefore generate varying word embeddings depending on the context of a word in the given sentence. For example, in the sentence "he went to the bank by the river bank", the word "bank" in the first half of the sentence would represent a financial institution, whereas in the latter half of the sentence would represent the edge of a river.

In contrast to Glove embeddings, which are produced by a word-based model, ELMo embeddings are produced by character-based models and use character convolutions. ELMo embeddings are context-dependent because their design is based on underlying bi-directional LSTM language models. Additionally, since ELMo embeddings are character-based, ELMo can handle words that are beyond the training vocabulary.

Finally, BERT further extends the bi-directional nature of ELMo by training on sub-words, thus striking a good balance between characters and word-based representations. This also allows BERT to handle out-of-vocabulary cases, which word-based models suffer from. Additionally, BERT is trained on an underlying transformer architecture, which has proved superior than LSTMs. BERT uses a masked language model where it arbitrarily masks a certain percentage of words in a sentence during training and learns these representations, thus allowing BERT to perform extremely well in context-based situations.

Table 2: Examples of Predictions

Test Sentence	**Original Rating**	**Prediction**	**AKB Entry**
The quarantine in Italy showed more clean waters in Venice with fish and a dolphin.	FALSE	FALSE	The canals in Venice have become cleaner after the shutdown, attracting schools of fish and dolphins.
Washing your hands helps prevent the spread of COVID-19.	TRUE	FALSE	Gargling salt water can protect against against COVID-19 by washing down the virus into the gut.
Chlorine dioxide cures COVID-19.	FALSE	FALSE	Chlorine dioxide, or Miracle Mineral Solution (MMS) can cure Covid-19.
Garlic can cure patients infected with the coronavirus	FALSE	FALSE	Coronavirus can be cured by sniffing clove and camphor and by drinking water, the virus will go to the stomach and the acid in the stomach will kill the virus.

5 Experimental Results

5.1 Setup

In this section, we evaluate our model through different experimental results. Our experimental setup is as follows. We have an AKB which contains 4139 factual mistakes. We have also collected 50 truthful claims. We select 50 random entries from the AKB along with the 50 truthful claims, we create a test set, consisting of 100 entries. Hence, the AKB used for this experiment will consist of $n = 4139 - 50 = 4089$ entries. Our main goal is to measure precision and recall of our system by attempting to classify the claims in the test set as factual mistakes. In this experiment we use $k = [5, 10, 20]$ as part of the top k most similar AKB entries given a test claim (Section 4.3). We report performance for these different values of k, in terms of precision, recall and F-1 score.

As stated in Section 4, our model classifies a test claim c as a factual mistake if $e_j \rightarrow c$ for $1 \leq j \leq n$ but makes no classification attempt on claims where $e_j \not\rightarrow c$.

Let a be the number of factual mistakes in our test set (50 in this case). Let b be the total number of claims, which were classified as factual mistakes from our model. Let c be the number of claims which were classified as factual mistakes from our model **and** have an initial rating of “FALSE”.

We measure precision as c/b, where c is the number of correct classifications and b is the total number of classifications. We measure recall as c/a, which is the number of correct classifications over the total number of mistakes in the test set (a).

5.2 Analysis

In our experiments we use different methods to extract sentence embeddings from the AKB and test set. For each entry in the AKB and test set we extract sentence embeddings by making use of different methods, like GloVe, RoBERTa, ELMo and USE. As stated above, by extracting sentence embeddings from the entries, we capture their semantic meaning and hence we can approximate the similarity between AKB and test sentences. As described in Section 4.3, for each entry in the test set, we select the top k most similar AKB ones, where similarity is measured as the cosine of the vectors in the d-dimensional space. Having a good quality of contextual vectors is a really important piece of our method, as for a given test sentence, the top k AKB premises need to be relevant to it, such that our NLI model can predict entailment, if the test sentence constitutes a factual mistake.

The need for accurate contextual vector representation motivated us to use many different models. It is worth noting that USE is the only encoder that yields sentence embeddings, which is contrary to the other models, which produce word embeddings. This means that for GloVe, RoBERTa and ELMo, the sentence embeddings for the AKB and test entries are calculated as the mean vector of the words in the sentence (i.e mean-pooling). By doing that, we transform a $n \times d$ matrix into a d dimensional row vector, where n is the number of words in the sentence. This operation distorts some of the original context, but is still deemed as a representative representation of the whole context of the sentence. We can see from Figure 1 that the recall of USE is larger that other approaches, while the precision is also among the highest. As a result the F-1 score reported is the highest for both values of k, which means that the yielded sentence embeddings capture the context adequately allowing us to correctly approximate similarity in the d-dimensional space.

As stated in Section 4.3 RoBERTa and ELMo constiture SOTA models that compute word embeddings, which depend on the neighbouring context in a bidirectional fashion. Also both of these models solve the Out-Of-Vocabulary problem, because they use character embeddings (ELMo) and sub-word information (BERT). Hence, we expected their recall performance in our similarity task to be better than reported in Figure 1.

Furthermore, our decision of using the top k most similar AKB entries as a set of premises and not the whole AKB turned out to benefit the performance of our approach as far as precision is concerned. Using the top k similarity ap-

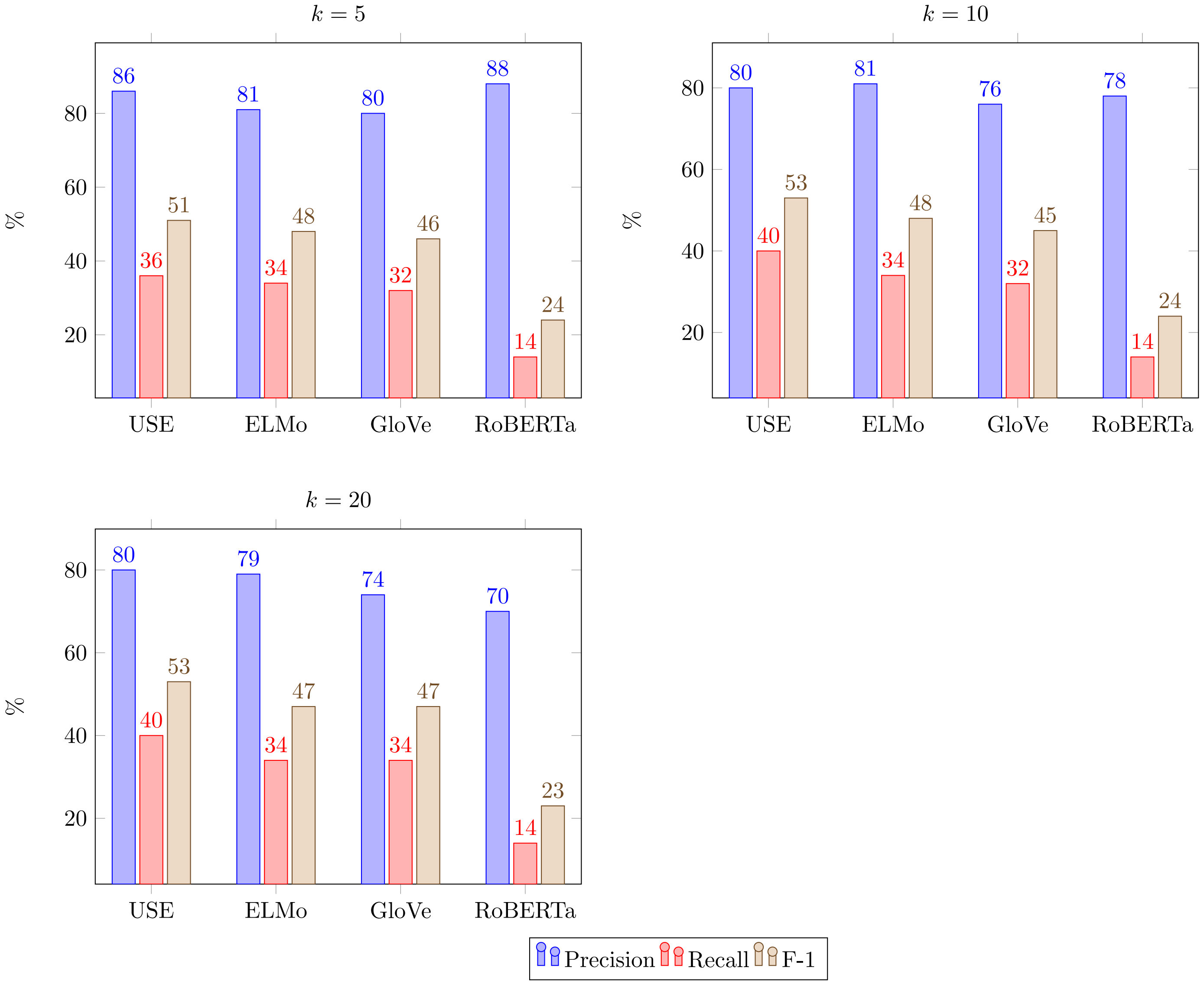

Figure 1: Precision, Recall and F-1 Score for different models with and values of k.

proach not only did it make our method salable and practical in terms of runtime and resource consumption, but it also increased the precision of RoBERTa in the NLI task, as we avoid trying to predict entailment for sentence pairs that do not have any contextual similarity. To be more precise, tackling NLI is a very tough task, and by limiting the number of unrelated sentence pairs (i.e AKB and test entries), we "help" the model achieve higher precision. This can be clearly shown in Figure 1, where the average precision is 84% for $k = 5$ in contrast to a precision of 79% and 76% for $k = 10$ and $k = 20$ respectively. This is an indication of an inversely proportional relationship between k and the precision.

We can see that all of the approaches achieve high performance but struggle in terms of recall. As discussed above, low recall was expected, since is inherent to the nature of our dataset. Having a dataset of approximately 4100 entries does not allow any model to cover a wide range of sentences related to COVID. This was also noticed in the web interface, as possible number of statements about the virus is much larger than the size of our AKB. Thus, the low recall can be attributed to the lack of a large dataset and could be solved if one collects more False sentences about the Coronavirus. We expect that a larger dataset with the existing method, will be able to adequately cover a wide variety of claims while retaining the current high precision.

Finally in Table 2, we show some examples of our Experimental Results. We show examples of correct and incorrect predictions of our model. We can see that our model can generalize quite well as the AKB entries which entail the Test sentences are different in structure and tone but similar in meaning. In the correct predictions we are able to classify a test sentence as incorrect and give an AKB entry as a justification of our prediction. Mistakes occur when we classify a test sentence as incorrect, when it's original rating is "TRUE".

6 Conclusion

In this project we tried to tackle a tough but very important problem that our society faces today, due to the large amount of misinformation concerning the COVID-19. We proposed a novel method, utilizing SOTA Language Models in the Natural Language Inference task and the concept of an Anti Knowledge Base. We managed to show that even by having a relatively small size of an AKB, automated Fact Checking of COVID-19 is possible and our proposed approach achieves very high performance with a maximum precision of 88%. We think that the results are significant in terms of sufficiently preventing the misinformation about COVID. We are also happy to provide a web interface where our method is readily available for everyone to use and Fact Check any spurious claims that they came across the Web.

References

[1] Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*, 2015.

[2] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Georgios Karagiannis, Immanuel Trummer, Saehan Jo, Shubham Khandelwal, Xuezhi Wang, and Cong Yu. Mining an" anti-knowledge base" from wikipedia updates with applications to fact checking and beyond. *Proceedings of the VLDB Endowment*, 13(4):561–573, 2019.

[5] 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. *arXiv preprint arXiv:1907.11692*, 2019.

[6] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

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

[8] Adina Williams, Nikita Nangia, and Samuel R Bowman. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1704.05426*, 2017.

[9] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27, 2015.