QUESTION ANSWERING

Name : GAURANGA DAS
Student ID : a1910652

The University of Adelaide

COMP SCI 7417 Applied Natural Language Processing
Lecturer: Dr. Alfred Krzywicki

Table of Contents

Abstract 2

1. Introduction 2

2. Preprocessing 3

3. System Architecture 3

3.1. Finding the most relevant sentence 4

3.2. Answering the question 4

3.3. Coreference Resolution 4

4. Model Selection and Training 5

4.1. Model Selection 5

4.2. Evaluation Metrics 5

4.3. Implementation Details 6

4.4. Results 6

5. User Interaction 6

6. System Evaluation 7

7. Conclusion 12

8. References 13

9. Appendix 14

A. MobileBERT 14

B. Sentence-BERT 15

Abstract

The aim of the project is to develop a question answering system that can extract an answer from an article given a query and the article number from the user. The overall question-answering system is divided into two parts:- 1. Extracting the most relevant sentence from an article and 2. Finding the answer snippet from the most relevant sentence. For finding the most relevant sentence, a deep learning model called Sentence-BERT (Reimers & Gurevych 2019; Aarsen 2022) was used and then the confidence score for that sentence was also calculated using cosine similarity (Natchiappan 2019). For finding the answer snippet, a deep learning pre-trained model called MobileBERT (Sun et al. 2020; Shamporov 2024) which is a variant of the original BERT (Devlin et al. 2019) model and has been fine-tuned on the SQuADv2.0 (Rajpurkar, Jia & Liang 2018) dataset was used. It was observed that the overall question answering system did not perform very well. When the most relevant sentence was extracted correctly, the MobileBERT model answered almost every question correctly. However, the Sentence-BERT model made quite a lot of mistakes in extracting the most relevant sentence. It can be inferred that to improve the system, a better model is needed to find the most relevant sentence. The MobileBERT model was fine-tuned on the SQuADv2.0 dataset which only contains factual questions. This means that the model is also limited to answering similar types of questions. Due to limited compute resources, a larger model could not be trained. Coreference resolution (Otmazgin 2023) is also applied to the article. Doing so led to an improvement in performance. In the text below context refers to both a paragraph and sentence. During fine-tuning, the context is a paragraph from the SQuADv2.0 dataset and during testing, the context is the most relevant sentence.

1. Introduction

Question answering has become increasingly important in the digital world, as it allows users to interact with computer systems as though they are talking to a human being. These systems can process large amounts of text, including news articles, and provide answers to questions given by the user. This enables people to save valuable time by getting the answer directly without the need to read through an entire article. More advanced systems based on language models can provide additional information such as summarizing a big article, finding relevant news articles for a user etc.

For businesses, leveraging these question answering systems in customer service centers can be very advantageous. Instead of having a human answer a customer's queries, a well-trained system can reply much quicker and more accurately. This is also beneficial for financial companies that rely on large amounts of data to make vital decisions. News media outlets could also use these systems to enable the user to extract information much quickly, rather than them having to read the entire article. Recently, there has been a surge of legal firms using question-answering systems to quickly get answers from vast amounts of text.

The system is an extractive question answering system. This means that it extracts an answer span from a context given a user question. If the correct answer is present in the context then the question answering system can predict the correct answer. However, if the correct answer is not present in the context and instead needs to be generated by the model then the system cannot give the correct answer. This is because the model is not a generative model and therefore, cannot create new information, instead it predicts the start and end probability of the answer span in the context. Another limitation is that the model can extract only a single span from the context. If the correct answer requires extraction of multiple spans from the context, the model can provide only one of the correct spans. While testing it was observed that the Sentence-BERT (Reimers & Gurevych 2019) model responsible for extracting the most relevant sentence was not always reliable. If the Sentence-BERT model extracts the wrong sentence then, the MobileBERT (Sun et al. 2020) model has no chance of predicting the correct answer. The SQuAD (Rajpurkar, Jia & Liang 2018) dataset the model was fine-tuned on has questions that are fact-based. There are no questions that require logical reasoning or mathematical calculations so, the model cannot answer those types of questions.

The project requires multiple tasks to be completed. The SQuADv2.0 dataset is downloaded and processed using the preprocessing techniques outlined below. Then, the MobileBERT model is fine-tuned on the dataset. After that, the user enters the article number and queries and corresponding answers. For testing, the most relevant sentence must be identified from the article for each query and its confidence score must also be calculated. Next, from the most relevant sentence and the question the answer snippet must be extracted. Coreference resolution (Otmazgin 2023) should also be applied.

2. Preprocessing

For this project the following preprocessing steps are applied:-

i. Conversion of text to lowercase
ii. Unnecessary whitespaces such as extra whitespaces between words, trailing or leading whitespaces are removed (Fredtantini 2014)
iii. Normalization of text is done so that any character that is non-ASCII is converted to its ASCII equivalent character. For eg., ‘Café Münchën’ is converted into ‘Cafe Munchen’ (MiniQuark 2009)

The MobileBERT model from HuggingFace comes with a MobileBERT tokenizer (Shamporov 2024) which performs WordPiece tokenization algorithm to convert the text to individual tokens. In addition to the above preprocessing steps special tokens such as the `[CLS]` token is added at the start of every input sequence and `[SEP]` token is added to separate the context/article and question and to the end of every input sequence.

3. System Architecture

The entire system is divided into three main components:- finding the most relevant sentence, answering the question from the most relevant sentence and applying coreference resolution. First, given a user question the most relevant sentence is extracted from the article. Then, the most relevant sentence and the question is used to find the answer snippet. Finally, coreference resolution is applied to the article and the steps above are repeated again. The details of each are described below.

3.1 Finding the most relevant sentence

To find the most relevant sentence from the article a pre-trained deep learning model called Sentence-BERT (Reimers & Gurevych 2019) was used from HuggingFace (Aarsen 2022). The Sentence-BERT model generates sentence embeddings for the question and for each sentence in the article. First, the article is split into individual sentences using the NLTK library (Slider 2016). Then the question and each sentence is encoded which produces a vector representation for each of them. Finally, the cosine similarity between the question vector and each sentence's vector is computed (Natchiappan 2019). The sentence with the highest cosine similarity score is chosen as the most relevant sentence. The cosine similarity score is the confidence score for that sentence and is also returned. This is because the cosine similarity gives a value between 0 and 1. A value close to 1 indicates that two vectors are similar and a value close to 0 indicates that two vectors are dissimilar. Therefore, the cosine similarity between the question vector and the sentence vector is an estimate of the confidence probability that a sentence is the most relevant sentence.

3.2 Answering the question

After the most relevant sentence has been extracted the question answering is performed using the MobileBERT (Sun et al. 2020) model which is a variant of the original BERT (Devlin et al. 2019) model. The reasons for choosing MobileBERT are mentioned in the next section. The pre-trained MobileBERT model and its weights are obtained from Huggingface (Shamporov 2024). They also provided a tokenizer for the model that can be used to convert a piece of text into numerical tokens and provide access to special tokens such as end-of-sentence token and separator token. The question and the context goes through various preprocessing techniques outlined in the Preprocessing section. Then both the context and the query are joined together to form one string but separated by a special `[SEP]` token. We then add the `[CLS]` token at the beginning and the `[SEP]` token at the end. The final input text is `[CLS] Sentence [SEP] Query [SEP]`. This input text is converted to numerical tokens using the tokenizer and then passed to the model. The model outputs are then used to calculate the start and end probabilities for each input token. Using this we extract the answer from the sentence. One thing to note is that if the model predicts either the start and or index of the answer span as 0, it means that the model thinks that the sentence doesn't contain the answer to the question. In such a case, 'NO ANSWER' will be predicted.

3.3 Coreference Resolution

The question-answering system also includes a coreference resolution component that was applied to the article to help the model to discern which entity a phrase is referring to. For implementation, the *fastcoref* (Otmazgin 2023) Python package was used, which is available as an open-source tool. The popular NLP module *spacy* was also used to integrate the *fastcoref* package into *spacy*'s pipeline and then coreference resolution was applied on the article using the *spacy* module.

4. Model Selection and Training

4.1 Model Selection

The model used for question answering is MobileBERT (Sun et al. 2020). This model is a variant of the original BERT (Devlin et al. 2019) model. The BERT model was pre-trained on large amounts of text and achieved state-of-the-art results on various tasks through fine-tuning and minimal training. However, it is very large and has a high inference time especially when not using a GPU. The base model of the BERT has 110 million parameters which makes it very difficult to train. MobileBERT on the other hand has only 25 million parameters and achieves performance comparable to the original BERT model. In fact, MobileBERT outperformed the BERT model in question answering on the SQuAD (Rajpurkar, Jia & Liang 2018) dataset. This means it can be trained quicker and has a lower inference time than BERT. Therefore, for developing a question answering system using MobileBERT is an excellent choice. The MobileBERT model is explained in detail in Appendix Chapter A.

For finding the most relevant sentence, the Sentence-BERT (Reimers & Gurevych 2019) model was used. Given any two sentences, Sentence-BERT can calculate the similarity between them. The idea is that if a sentence contains the answer to a question it will be similar to the question and have high cosine similarity. Since, the model uses word embeddings it means that the model can detect similarity even if a sentence and the question use different phrases. The Sentence-BERT model is explained in detail in Appendix Chapter B.

4.2 Evaluation Metrics

For evaluating the model during fine-tuning, the loss and F1 score is measured (Kim 2018). The loss function measures how far off the model's performance is. A decreasing loss indicates that a model is actually learning something. It can also identify if the model is overfitting. The loss can tell if the model is accurately predicting the answer. F1 score is also chosen because it considers both the precision and recall. Precision measures the proportion of the predicted answer provided by the model that was actually correct. Recall measures the proportion of the actual correct answer the model actually found. Both these metrics are important as they indicate the accuracy of the question-answering system. Therefore, F1 score is chosen because it considers both the precision and recall and indicates the quality of the answer predicted by the model. F1 score is only measured on the development dataset and is used to choose which model will be used for testing.

4.3 Implementation Details

For fine-tuning the SQuADv2.0 dataset is chosen, as it is one of the best open-source datasets available online for training question-answering systems. This dataset contains 130,000 question-answer pairs. Not all the questions in the dataset are answerable therefore, forcing the model to identify if the answer to question exists in a given context or not. To reduce the training time, only a subset of the question-answer pairs were chosen for training. The model was trained using the multi-class cross-entropy loss function and the Adam optimizer. While fine-tuning the gradients of the model became very large which is known as the exploding gradients problem. Therefore, gradient clipping (Metsai 2016) was applied to prevent the gradients from becoming too large. The data was passed to the model in batches of size 32. For each batch the loss and F1 score was calculated and the final result was obtained by taking the mean of each. The code was implemented using the Pytorch library.

4.4 Results

The fine-tuning results on the training and validation set is displayed below

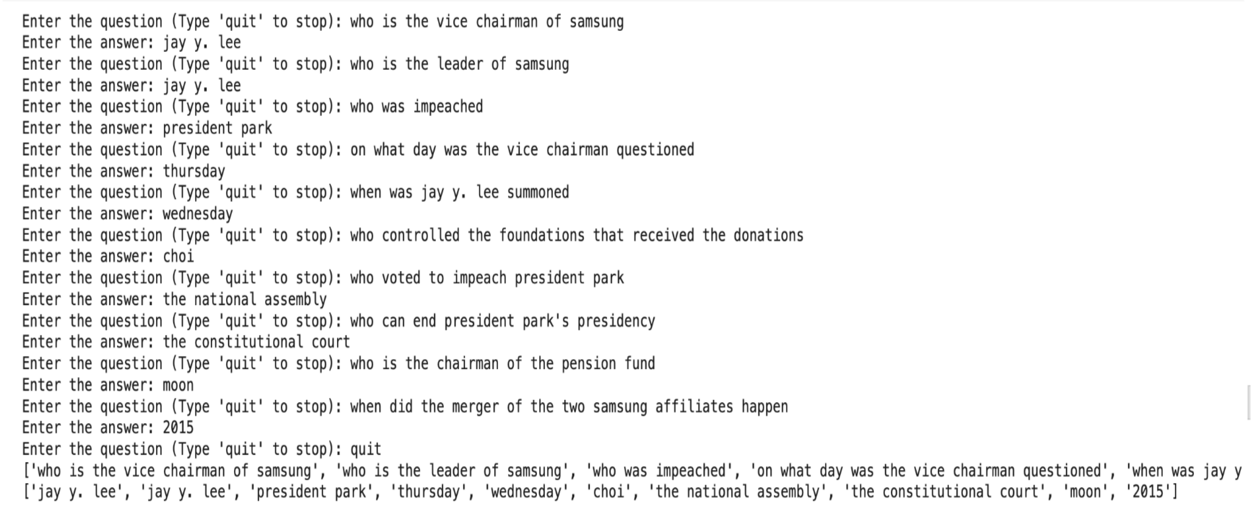
Epoch	Train Loss	Valid Loss	Valid F1
1	26106.10	2.04	68.93%
2	1.87	1.81	73.98%
3	1.49	1.81	75.32%
4	1.21	1.81	76.14%

The model was not trained further than four epochs because it was clear that the model started to overfit the training data by the fourth epoch. For testing the model from the fourth epoch was chosen because it had the highest validation F1 score. It suggests that the model was best at accurately predicting answers on unseen data.

5. User Interaction

To interact with the system simply run all the cells. There is a 'TRAIN_MODEL' variable in the 5th code cell that decides whether to fine-tune the model on the dataset or not. This is because the entire process of preprocessing the dataset and fine-tuning the model can take a lot of time. Therefore, the model weights have been uploaded to Github after fine-tuning and can be downloaded in the notebook. If the 'TRAIN_MODEL' variable is set to True, the SQuADv2.0 (Rajpurkar, Jia & Liang 2018) dataset will be downloaded and then the model will be fine-tuned on the dataset. By default, the 'TRAIN_MODEL' variable is set to False which means that the fine-tuning and data preprocessing will not take place instead, the model weights will be

downloaded and will be used for inference. Finally, a prompt will appear that will ask the article number. Type the article number and press enter and then another prompt will appear for the question. Type the question and press enter and then another prompt will appear where the answer has to be typed. After typing the answer press enter. Continue doing this until all questions and corresponding answers have been entered. Once done, simply type ‘quit’ when prompted to ask the question. Then for each question, the most relevant sentence, its confidence score and the predicted answer will be displayed. Finally, the F1 score for all the answers will be displayed. After that coreference resolution (Otmazgin 2023) is also applied on the article and then the process of finding the most relevant sentence and predicting the answer for each question is repeated again, this time with coreference resolution.
The image below shows an example of a user session. The user is prompted to type the question and then after pressing enter, the answer is typed and then enter is pressed again. This process is repeated until all questions and corresponding answers have been typed. Once done, type ‘quit’ when prompted the question to stop.



```
Enter the question (Type 'quit' to stop): who is the vice chairman of samsung
Enter the answer: jay y. lee
Enter the question (Type 'quit' to stop): who is the leader of samsung
Enter the answer: jay y. lee
Enter the question (Type 'quit' to stop): who was impeached
Enter the answer: president park
Enter the question (Type 'quit' to stop): on what day was the vice chairman questioned
Enter the answer: thursday
Enter the question (Type 'quit' to stop): when was jay y. lee summoned
Enter the answer: wednesday
Enter the question (Type 'quit' to stop): who controlled the foundations that received the donations
Enter the answer: choi
Enter the question (Type 'quit' to stop): who voted to impeach president park
Enter the answer: the national assembly
Enter the question (Type 'quit' to stop): who can end president park's presidency
Enter the answer: the constitutional court
Enter the question (Type 'quit' to stop): who is the chairman of the pension fund
Enter the answer: moon
Enter the question (Type 'quit' to stop): when did the merger of the two samsung affiliates happen
Enter the answer: 2015
Enter the question (Type 'quit' to stop): quit
['who is the vice chairman of samsung', 'who is the leader of samsung', 'who was impeached', 'on what day was the vice chairman questioned', 'when was jay y
['jay y. lee', 'jay y. lee', 'president park', 'thursday', 'wednesday', 'choi', 'the national assembly', 'the constitutional court', 'moon', '2015']
```

6. System Evaluation

For testing the question answering system, the following questions and answers were given. The article number ‘17574’ was chosen. For each test case, the predicted most relevant sentence, its confidence score and the predicted answer is also shown.

Question: `who is the vice chairman of samsung`
Actual Answer: `jay y. lee`
Most Relevant Sentence: `the de facto leader, jay y. lee, the vice chairman of samsung, will be questioned on thursday, according to the special prosecutor?s office, which recommended that he also be investigated on suspicion of perjury.`
Confidence Score: `62.71%`
Predicted Answer: `jay y . lee`

Most Relevant Sentence(with coreference resolution): the de facto leader, jay y. lee, the vice chairman of samsung, will be questioned on thursday, according to the special prosecutor?s office, which recommended that the de facto head of samsung also be investigated on suspicion of perjury.

Confidence Score: 65.27%

Predicted Answer(with coreference resolution): jay y . lee

Question: who is the leader of samsung

Actual Answer: jay y. lee

Most Relevant Sentence: mr. lee effectively runs samsung, south korea?s largest conglomerate he is the son of its chairman, lee who has been incapacitated with health problems.

Confidence Score: 67.50%

Predicted Answer: mr . lee

Most Relevant Sentence(with coreference resolution): the de facto head of samsung effectively runs samsung, south korea?s largest conglomerate the de facto head of samsung is the son of samsung's chairman, lee who has been incapacitated with health problems.

Confidence Score: 72.76%

Predicted Answer(with coreference resolution): lee

Question: who was impeached

Actual Answer: president park

Most Relevant Sentence: in its impeachment bill, the national assembly asserted that the donations were bribes, made with the expectation of political favors from the president.

Confidence Score: 45.37%

Predicted Answer: NO ANSWER

Most Relevant Sentence(with coreference resolution): the special prosecutor?s office, which recommended that he also be investigated on suspicion of perjury asked national assembly to file a perjury complaint against the de facto head of samsung, which would authorize a special prosecutor investigating the corruption scandal that led to president park ?s impeachment to open an investigation of that charge.

Confidence Score: 48.78%

Predicted Answer(with coreference resolution): president park

Question: on what day was the vice chairman questioned

Actual Answer: thursday

Most Relevant Sentence: the de facto leader, jay y. lee, the vice chairman of samsung, will be questioned on thursday, according to the special `prosecutor?s office, which recommended that he also be investigated on suspicion of perjury.`

Confidence Score: `56.04%`

Predicted Answer: `thursday`

Most Relevant Sentence(with coreference resolution): `the de facto leader, jay y. lee, the vice chairman of samsung, will be questioned on thursday, according to the special prosecutor?s office, which recommended that the de facto head of samsung also be investigated on suspicion of perjury.`

Confidence Score: `53.46%`

Predicted Answer(with coreference resolution): `thursday`

Question: `when was jay y. lee summoned`

Actual Answer: `wednesday`

Most Relevant Sentence: `the de facto leader, jay y. lee, the vice chairman of samsung, will be questioned on thursday, according to the special prosecutor?s office, which recommended that he also be investigated on suspicion of perjury.`

Confidence Score: `63.68%`

Predicted Answer: `NO ANSWER`

Most Relevant Sentence(with coreference resolution): `the de facto leader, jay y. lee, the vice chairman of samsung, will be questioned on thursday, according to the special prosecutor?s office, which recommended that the de facto head of samsung also be investigated on suspicion of perjury.`

Confidence Score: `58.91%`

Predicted Answer(with coreference resolution): `NO ANSWER`

Question: `who controlled the foundations that received the donations`

Actual Answer: `choi`

Most Relevant Sentence: `samsung gave the largest donations to ms. choi?s foundations, totaling $17 million.`

Confidence Score: `55.06%`

Predicted Answer: `samsung`

Most Relevant Sentence(with coreference resolution): `samsung gave the largest donations to ms. choi foundations, totaling $17 million.`

Confidence Score: `52.71%`

Predicted Answer(with coreference resolution): `samsung`

Question: `who voted to impeach president park`

Actual Answer: `the national assembly`

Most Relevant Sentence: `allegations that ms. park helped ms. choi extort millions in bribes from samsung and other companies are at the heart of` the corruption scandal that led to the national assembly?s vote to impeach her last month.

Confidence Score: 57.37%

Predicted Answer: the national assembly

Most Relevant Sentence(with coreference resolution): allegations that president park ?s helped choi a longtime friend of the president extort millions in bribes from samsung and other companies are at the heart of the corruption scandal that led to the national assembly?s vote to impeach president park ?s's last month.

Confidence Score: 59.54%

Predicted Answer(with coreference resolution): the national assembly

Question: who can end president park's presidency

Actual Answer: the constitutional court

Most Relevant Sentence: since then, ms. park?s powers have been suspended, and she is on trial at the constitutional court, which will ultimately decide whether to end her presidency.

Confidence Score: 66.68%

Predicted Answer: she is on trial at the constitutional court

Most Relevant Sentence(with coreference resolution): since then, president park ?s powers have been suspended, and president park ?s is on trial at the constitutional court, which will ultimately decide whether to end president park ?s's presidency.

Confidence Score: 75.41%

Predicted Answer(with coreference resolution): president park ? s is on trial at the constitutional court

Question: who is the chairman of the pension fund

Actual Answer: moon

Most Relevant Sentence: the national pension fund?s support was crucial for the merger, which analysts said helped mr. lee inherit control of samsung from his father.

Confidence Score: 41.74%

Predicted Answer: NO ANSWER

Most Relevant Sentence(with coreference resolution): a special prosecutor investigating the corruption scandal that led to president park ?s impeachment has been investigating whether samsung gave samsung's support to ms. choi in exchange for a decision by the national pension service to support a contentious merger of two samsung affiliates in 2015. moon chairman of the national pension service, was arrested last month on charges that moon chairman of the pension fund illegally pressured the national pension service to back a contentious merger of two samsung

affiliates in 2015 when moon chairman of the pension fund was south korea?s health and welfare minister.
Confidence Score: 37.84%
Predicted Answer(with coreference resolution): moon

Question: when did the merger of the two samsung affiliates happen
Actual Answer: 2015
Most Relevant Sentence: the national pension fund?s support was crucial for the merger, which analysts said helped mr. lee inherit control of samsung from his father.
Confidence Score: 65.27%
Predicted Answer: NO ANSWER
Most Relevant Sentence(with coreference resolution): the national pension service support was crucial for the merger, which analysts said helped mr. lee inherit control of samsung from mr. lee's father.
Confidence Score: 65.40%
Predicted Answer(with coreference resolution): NO ANSWER

Without applying coreference resolution the overall question-answering system achieved a F1 score of 41.17%. The overall question-answering system didn't perform very well due to the low F1 score. However, one thing to be noted is that the Sentence-BERT (Reimers & Gurevych 2019) model responsible for extracting the most relevant sentence made a lot of mistakes. In fact, when the correct relevant sentence was extracted the MobileBERT (Sun et al. 2020) predicted the correct answer most of the time. This means that the MobileBERT model was quite accurate in predicting answers based on questions from the news dataset even though it was never trained on it. This shows that the MobileBERT model is very powerful as it can generalize well to other datasets even if it is not trained on them. In fact, in the test cases looking at question 3,5,9 & 10 when the incorrect relevant sentence was extracted, the MobileBERT correctly predicted that the sentence does not contain the answer. When the correct relevant sentence was predicted the MobileBERT model was very accurate at predicting answers. This indicates that the only reason the overall question-answering system didn't perform well was because the Sentence-BERT model was not able to correctly extract the correct relevant sentence. One reason could be that the Sentence-BERT model was never trained to predict whether a sentence contains the answer to a question. In the paper (Reimers & Gurevych 2019), the researchers stated that they trained the model on Natural Language Inference (NLI) task and Sentence Textual Similarity (STS) task. Since, there is a mismatch between the tasks the model was trained on and the task it is being used for, the model is not able to perform well. Looking at the confidence scores, the Sentence-BERT model was not very confident in its predictions. This could be due to the aforementioned fact that the model was not trained to detect whether a sentence contains the answer to a question. The actual correct relevant sentence and the question would most probably be very different semantically even though they may share some common words and phrases.

After applying coreference resolution, the F1 score increased to 58.29%. This improvement in performance indicates that applying coreference resolution can lead to a significant improvement in performance. Looking at question 9 & 10, when coreference resolution was applied the Sentence-BERT model was able to correctly predict the most relevant sentence which enabled the MobileBERT model to predict the correct answer. Therefore, it can be said that applying coreference resolution can lead to some improvement in performance.

7. Conclusion

The MobileBERT (Sun et al. 2020) model achieved a good F1 score on the SQuADv2.0 (Rajpurkar, Jia & Liang 2018) dataset. However, the overall question-answering system(including the Sentence-BERT model) achieved a much worse performance on the provided test cases. The Sentence-BERT (Reimers & Gurevych 2019) model is mostly responsible for the poor performance because when it predicted the correct relevant sentence the MobileBERT model was able to predict the correct answer with a good accuracy. But, when it incorrectly predicted the most relevant sentence the MobileBERT model had no chance to predict the correct answer. Therefore, it can be inferred that using the Sentence-BERT model for extracting the most relevant sentence is not a very good approach for question-answering. The MobileBERT model on the other hand was able to predict the correct answer span when the correct relevant sentence was given even though the model was trained on the SQuADv2.0 dataset, it was still able to answer questions from the News dataset. It can be deduced that the MobileBERT (Sun et al. 2020) model can achieve satisfactory results in question-answering and is able to understand the language semantics. The coreference resolution (Otmazgin 2023) model improved the F1 score by helping the Sentence-BERT model to correctly predict the most relevant sentence in some test cases. Even though the performance improvement was marginal, it can be concluded that coreference resolution can help to improve performance in question-answering.

During fine-tuning, one big problem was that it took a lot of time to train the model and process the dataset. Therefore, the model weights have been uploaded and available to download from Github. Another problem encountered during fine-tuning was that the gradients of the model became too big which led to the loss becoming ‘nan’. This is known as the exploding gradients problem. To rectify this, gradient clipping (Metsai 2016) was applied.

According to the results above, the primary reason for degradation in performance during testing was due to the fact that the Sentence-BERT model was not able to correctly predict the relevant sentence. Therefore, to find the most relevant sentence more accurately the Sentence-BERT model could be fine-tuned on the task of predicting whether a sentence contains the answer to a question or a generative model like GPT-3 could be used. To get better results on question-answering, a bigger model(such as $\text{BERT}_{\text{LARGE}}$) could be trained for more epochs instead of using MobileBERT. The MobileBERT (Sun et al. 2020) model used can only extract an answer from an article if the exact answer is present in the article. A generative model could also be used (such as GPT-3) for further improvements because it can extract multiple spans and even generate new data to answer the questions in case the exact answer is not present in the context. The SQuAD dataset is very limited in scope because it contains only fact-based questions and answers. A better dataset is also needed where answering questions requires generating new information, reasoning or basic algebra instead of simple fact-based questions.

8. References

Aarsen, T 2022, *all-MiniLM-L6-v2*, HuggingFace, https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Devlin, J, Chang, M-W, Lee, K & Toutanova, K 2019, 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding', *Proceedings of NAACL-HLT 2019*, pp. 4171-4186, https://aclanthology.org/N19-1423.pdf

Fredtantini 2014, *python - remove whitespaces that are not needed [duplicate]*, StackOverflow, https://stackoverflow.com/questions/27618462/python-remove-whitespaces-that-are-not-needed

Gugger, S, Zucker, A, Jik, L 2024, *transformers/src/transformers/models/mobilebert/tokenization_mobilebert.py*, Github, https://github.com/huggingface/transformers/blob/main/src/transformers/models/mobilebert/tokenization_mobilebert.py

Kim, T, 2018, *BiDAF-pytorch/evaluate.py*, Github, https://github.com/galsang/BiDAF-pytorch/blob/master/evaluate.py

Metsai, A 2016, *Pytorch: test loss becoming nan after some iteration*, StackOverflow, https://stackoverflow.com/questions/66648432/pytorch-test-loss-becoming-nan-after-some-iteration

MiniQuark 2009, *What is the best way to remove accents (normalize) in a Python unicode string?*, StackOverflow, https://stackoverflow.com/questions/517923/what-is-the-best-way-to-remove-accents-normalize-in-a-python-unicode-string

Natchiappan, V 2019, *Using sklearn how do I calculate the tf-idf cosine similarity between documents and a query?*, StackOverflow, https://stackoverflow.com/questions/55677314/using-sklearn-how-do-i-calculate-the-tf-idf-cosine-similarity-between-documents

Otmazgin, S 2023, *fastcoref*, Github, https://github.com/shon-otmazgin/fastcoref

Rajpurkar, P, Jia, R & Liang, P 2018, *SQuAD 2.0(Stanford Question Answering Dataset)*, https://rajpurkar.github.io/SQuAD-explorer/

Reimers, N, Gurevych, I 2019, ‘Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks’, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pp. 3982-3992, https://aclanthology.org/D19-1410.pdf

Shamporov, V 2024, *MobileBERT*, HuggingFace, https://huggingface.co/docs/transformers/model_doc/mobilebert

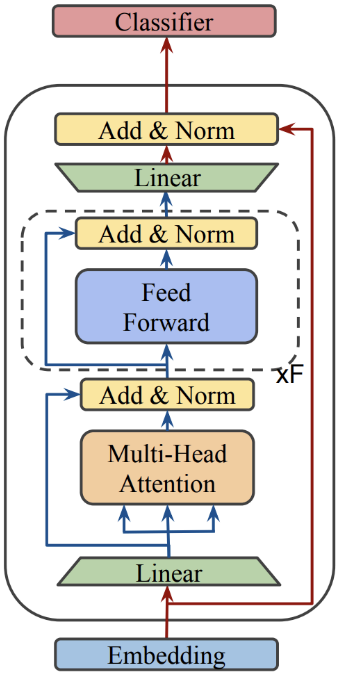
Slider 2016, *Tokenize a paragraph into sentence and then into words in NLTK*, StackOverflow, https://stackoverflow.com/questions/37605710/tokenize-a-paragraph-into-sentence-and-then-into-words-in-nltk

Sun, Z, Yu, H, Song, X, Liu, R, Yang, Y & Zhou, D 2020, ‘MobileBERT: a Compact Task-Agnostic BERT for Resource-Limited Devices’, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2158-2170, https://aclanthology.org/2020.acl-main.195.pdf

Trevett, B 2023, *pytorch-sentiment-analysis*, Github, https://github.com/bentrevett/pytorch-sentiment-analysis

9. Appendix

A MobileBERT



The architecture of the MobileBERT (Sun et al. 2020) model is very similar to the original BERT (Devlin et al. 2019) model but it is much smaller. Its depth(number of Transformer encoder layers) is the same as the original BERT, but each layer has a smaller number of parameters. One key difference is that each Transformer encoder layer of MobileBERT has a linear/fully-connected layer at the beginning and end to adjust its input and output dimensions. Another difference is that each layer of MobileBERT has multiple feed-forward networks to ensure the ratio of parameters in the Multi-Head Self-Attention layer and the Feed-Forward networks layers is the same as the original BERT. A technique called knowledge distillation is used where a larger model called the teacher network is trained first which is then used to train the MobileBERT model. This teacher network has the same architecture as the MobileBERT but it has more parameters(comparable to the original BERT model). Before training MobileBERT, the teacher network was trained using the same objectives as the original BERT model. For training MobileBERT, the following three objectives were used:-

1. **Feature Map Transfer**

The output feature maps of the teacher model and the MobileBERT model at each layer are used to calculate the mean squared error to transfer layer-wise knowledge from the teacher to MobileBERT

2. **Attention Transfer**
   The per-head self-attention distributions of MobileBERT and the teacher model is used to calculate the KL-divergence loss to help the MobileBERT to learn to properly attend to the input sequence like the teacher model
3. **Pre-training Distillation**
   This combines the masked language modeling loss and next sentence prediction loss which is similar to the original BERT model.

Using the three objectives, different combinations can be used for training. The paper discusses three different strategies however, here only the best performing one will be discussed. Progressive Knowledge Transfer (PKT) works by progressively training each Transformer encoder layer separately. First, the lowest layer is trained using the Feature Map Transfer and Attention Transfer objectives and then progressively each of the layers is trained using the same objectives until the final layer is trained. Finally, the Pre-training distillation objective is used to train the MobileBERT model. For training, the masked language modeling and next sentence prediction was used. The exact implementation details are in the paper.

B Sentence-BERT

The Sentence-BERT (Reimers & Gurevych 2019) model is a pre-trained model to generate sentence embeddings that can be used to compare sentences using cosine similarity. Sentence-BERT uses the same architecture as the original BERT (Devlin et al. 2019) model. To get sentence embedding, the input tokens of a sentence are passed through the BERT model and then the mean of all the token output vectors is calculated to derive the sentence embedding vector for the input token sequence. The model is trained using two objective functions:-

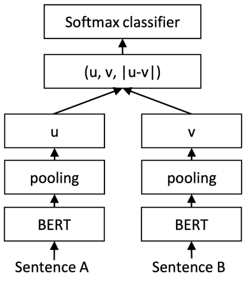


Figure 1: SBERT architecture with classification objective function, e.g., for fine-tuning on SNLI dataset. The two BERT networks have tied weights (siamese network structure).

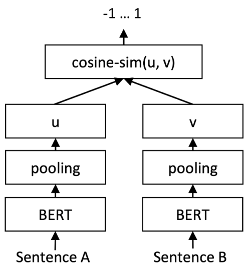


Figure 2: SBERT architecture at inference, for example, to compute similarity scores. This architecture is also used with the regression objective function.

1. **Regression Objective Function**
   Two input sentences are passed to the Sentence-BERT model separately and the sentence embedding for each of them is obtained. Then, the cosine similarity between them is calculated and the mean-squared error loss is used as the objective function.
2. **Classification Objective Function**

Two input sentences are passed to the Sentence-BERT model separately and the sentence embedding for each of them is obtained. The sentence embeddings are concatenated along with their element-wise difference and then the result is multiplied by a trainable weight to produce a vector whose length is the no. of classes which is then used to train the model using cross-entropy loss.

Furthermore, the model is fine-tuned on two tasks:-

1. **Natural Language Inference (NLI) Task**
   The model is given a premise sentence and it has to predict whether a second sentence is True, False or undetermined. This trains the model to understand the language semantics and basic reasoning skills. For this task the classification objective function is used.
2. **Semantic Textual Similarity (STS) Task**
   The model is given two sentences and it has to predict a degree of similarity between those sentences by giving a score from 0 to 5. This helps the model to determine whether two sentences are similar or not. For this task the regression objective function is used.

The resulting model after training on these tasks is called Sentence-BERT. Given two sentences it can compute the similarity between them using cosine similarity.