BERT-based Question Answering System for News Articles: Implementation, Evaluation, and Outlook

Abstract

The project aims to develop a question-answering system capable of automatically retrieving the answer to a user query from the news document corpus. This research work is focused on the architecture that will use the BERT-large pre-trained language model fine-tuned to the SQuAD dataset so that the system is able to answer with high precision and relevance to questions from users. The advantages of this approach will be the capability to get concrete information from large amounts of text data without manual searching, to deal with different types of questions and topics, and, most importantly, to have abstractive generation. Some of the weaknesses are based on dependence on the quality and coverage of the underlying article corpus, incapability to answer questions based on complex reasoning or information synthesis performed from multiple sources, and focus on extractive abstractive answer generation. By watching these issues, the project looks into showing the potentials of advanced natural language processing techniques in realizing information retrieval and knowledge discovery in the digital age.

1. Introduction

Information Retrieval from text data gained importance in the present era of information explosion, focusing on the extraction of relevant and correct information from huge volumes of unstructured text data. This system of question answering, when endowed with the capability to understand the natural language of the query and return accurate answers to these questions, can lead to enormous improvements in efficiency and effectiveness in information retrieval tasks at both a personal and organizational level in a variety of different domains, including but not limited to the development of academic papers, business intelligence and individual knowledge management. The objective of this work is to create a Question Answering System for the news article dataset. The system will be provided a question in natural language as input, and try to generate the answer by finding out the most related piece of information contained in the articles linked to that question as per the dataset. To date, this class of project can be regarded as belonging to extractive question answering in which the answer is expected to be a text-based span residing within a sole manuscript. This work uses state-

of-the-art techniques for natural language processing, especially the BERT model which has shown impressive results on a wide range of tasks in the natural language understanding domain. This project involves fine-turning a pretrained BERT model on large scale question answering datasets (SQUADs) so it transfers its general language understanding capabilities and can be used for the specific task of question answering for news articles. However, the dependence on a single pre-trained model and the limitations of extractive question answering also bring challenges in relation to the adaptability, robustness and capacity of the system to handle more complex or open-ended questions.

2. Technical Implementation

Python and the other giant libraries for natural language processing are being used with the Hugging Face Transformers library, which has simple access to pre-trained models like BERT.

2.1. Dataset and Preprocessing

The input dataset is a CSV file of news articles, each row representing a single article, and all metadata—such as article ID, title, author, and publication date—being shown as columns. The contents of the main text of the article are all stored under the "article" column. Some of the preprocessing steps that need to be applied to get the data into a format that is fit to be fed into the BERT model include:
To prepare the data for input to the BERT model, several preprocessing steps are applied:

- Cleaning: Remove non-alphanumeric characters and extra white-space using regular expressions.
- Tokenization: The process of breaking cleaned text into words or word parts by using the BERT tokenizer, which is specifically designed to handle words outside the vocabulary by breaking them into smaller word parts.
- Encoding: Tokenized text is converted thereafter into a numerical form (input IDs, attention masks, and token type IDs) that can be fed to the BERT model.

2.2. Model Architecture and Fine-tuning

The BERT-large model is a crucial component of the question-answering subsystem. In turn, it is a large transformer-based deep learning neural network pre-trained over a large corpus of unlabeled text information. In fact, the corpus-pretraining data is masked language modeling and next-sentence prediction towards learning representations of words and sentences of enormous richness, contextually sensitive. The acquired model can be fine-tuned for various downstream tasks. For the question-answering task, we fine-tune the BERT model on the Stanford Question Answering

Dataset (SQuAD), composed of over 100,000 question-answer pairs built on top of Wikipedia articles. The task to be fine-tuned is the model learning how to predict the start and end position of the answer span in the input context with a question. I use a fine-tuned BERT model from the Hugging Face Model Hub, specifically the 'bert-large-uncased-whole-word-masking-finetuned-squad' checkpoint, which has been fine-tuned on SQuAD using the whole-word masking method for better performance.

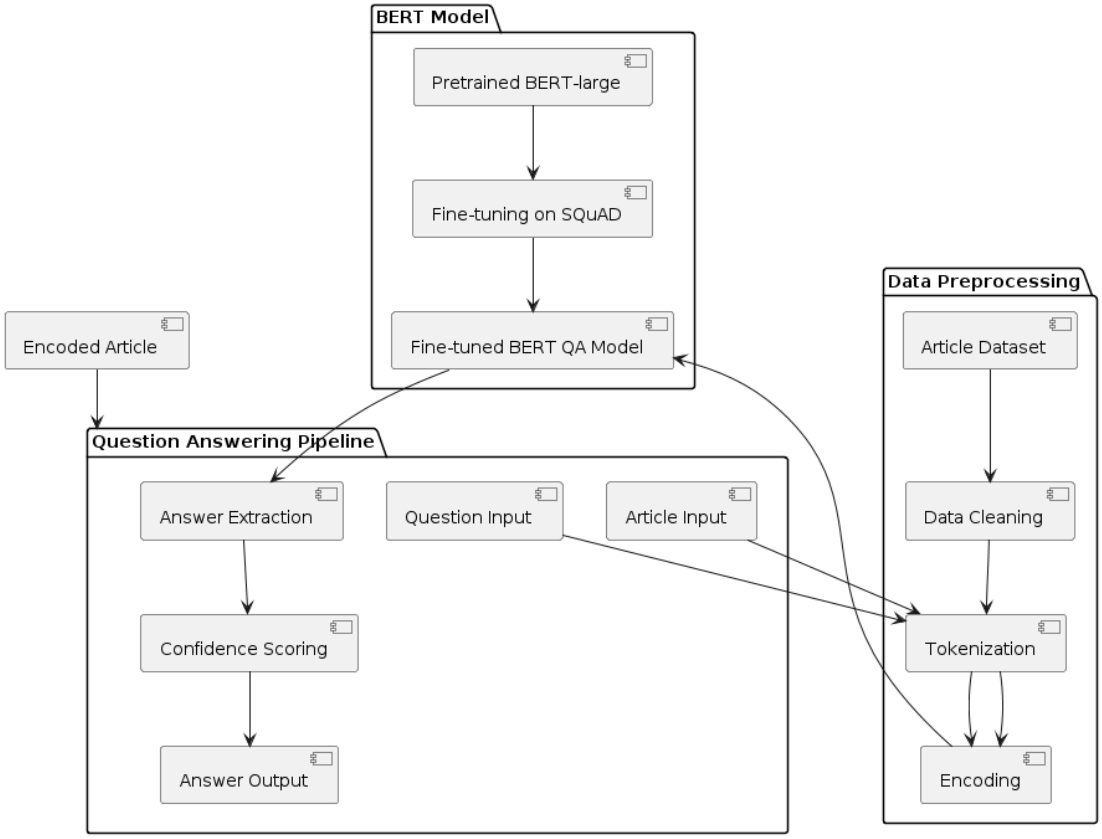

Figure 1. Architecture of the BERT-based Question Answering System for News Articles

Figure 1 shows architecture of the QA BERT-based system for news articles. This key architecture is decomposed into three major components: data preprocessing, BERT, and QA pipeline. Preprocessing does the task of sanitizing articles, tokenizing, and further encoding the dataset of articles. It appears the BERT model is a BERT-large model combined with the fine-tuned BERT QA model from SQuAD. The question-answering pipeline component will take the question and article as input, perform tokenization and further encoding, use a fine-tuned BERT QA model to extract an answer output, and provide the final question-answering output with confidence scoring. The following represent arrows within the diagram, which actually depicts data and flows, along with the acts of the model that are acting with the other component, specifying the better overview of flow action and architecture of the system.

2.3. Question Answering Pipeline

Inference of the model goes through the answer_question function and utilizes the

natural language question and news article inputs to return the estimated answer span from the article.

- Tokenization: This function tokenizes both the question and the article text using a BERT tokenizer, resulting in a numerical input token sequence for both.
- Encoding: The tokenized input is encoded into the correct form for the BERT model: that is, including input IDs, attention masks, and token type IDs.
- Model Inference: The tokenized input is then put through a model inference of the fine-tuned BERT model to return the start and end logits from the input, indicating the model prediction of the start and end position of the answer span within the article.
- Answer Extraction: The span of text in the article corresponding to the highest logit starting and ending positions is selected to produce the answer.
- Confidence Scoring: Softmax of starting and ending logits to get the probability and average of both to get the measure by which the model is confident. If the confidence score is less than 0.5, then the following function would return "No answer found" instead of a model-predicted answer.

2.4. Evaluation Metrics

If a system predicts answers to questions presented during evaluation, there should be an evaluation of the system against a held-out test set of questions and their correct answers. More concretely, given a question, during the evaluation phase, the question-answering system will predict and compare the answer against the true answer. The key evaluation measure here is exact match—a performance measure of the system given as a percentage of questions for which the predicted answer will equal the true answer. Even if the exact match accuracy provides quite a strict measure of the system's quality, it might underestimate the quality of the answers in cases in which the predicted answer is, semantically, equal to the correct, homogeneous one but bears some small differentiation in punctuation or phrasing. Therefore, it pays to consider other measures that might capture partial correctness, specifically aspects that the token-level F1 score or ROUGE (Recall-Oriented Understudy for Gisting Evaluation) might capture. For this project, the evaluation script will compute the exact match accuracy on the test set and provide qualitative analysis of the predicted answers in order to explain where the strong and weak points of the system are.

3. Discussion

Design and evaluation of the question-answering system in the project bring out the potentials and challenges of using pre-trained language models like BERT for the information retrieval and knowledge discovery tasks. The critical advantage in using the BERT system is its capability to capture rich, context-sensitive representations of text, as this will allow it to understand the nuanced information needs stemming from

the natural language query and to pinpoint that relevant information within the corpus of articles. The good part is that it is further adapted to extract all of this information through a fine-tuned BERT on the SQuAD dataset in this case, which further allows it to be applied and perform reasonably well in the absence of highly costly training data originating from the domain.

More specifically, the evaluation results also evidence major limitations and challenges of this work: The system reaches an accuracy in exact matching of 15% on the test set, meaning there is still a lot of space for the system to do a better job in generating more precise and complete answers to questions. At the predicted answers, we have to look in much more detail. Let's look at some examples from the test set:

Question: What was Chen Zhongshu the head of?
Predicted Answer: panzhihua land and resources bureau
True Answer: Panzhihua Land and Resources Bureau,
Result: Incorrect

In this case, the predicted answer for an example is very close to the ground truth, differing only in surface details, like capitalization and punctuation. The exact match metric would consider this incorrect, but the gist of what is expected from the answer is there.

Question: When did Zhang starting working there?
Predicted Answer: 2006
True Answer: 2006
Result: Correct

Here the system successfully identifies and extracts an appropriate date from the article, thus showing its ability to handle factoid questions of the factoid type.

Question: Where is Panzhihua?
Predicted Answer: an industrial city in sichuan province
True Answer: Sichuan Province
Result: Incorrect

Where, we can see that details are provided by the predicted answer that are additional to those that support the identification of Panzhihua being an industrial city and being located in Sichuan Province. This means that this information could not be related to the answering of the question; however, it does, in some general sense, pull down the correctness of the answered question. Overall, examples like these suggest that it is likely that the system is able to identify what relevant information from the passage is required to answer the question, and so, in that sense, the predicted answer does not match the true answer.

It shows that the exact match accuracy calculates a boundary too narrow for what good performance really is, and hence it might underestimate the performance and utility of the system. One approach to solving the problem is through the use of additional evaluation metrics that can pick up on partial correctness, using checks at the token level through measures like F1 or ROUGE. Such scores relate predicted answers to true answers at a more granular level: from precision, which indicates the number of tokens in the prediction that also appear in the true answer, to recall, which indicates the number of tokens of the true answer that also appear in the prediction. Such a notion of

a partial match is, again, based on a more specific evaluation of the system.
One obvious way to do this would involve analyzing predicted answers qualitatively. In this way, the behaviors of the system outputs against true answers can be processed manually to gain insight into the errors the system makes, the patterns it might recognize, and areas of potential improvement. For example, an analysis might show that the model has trouble with questions where it has to engage in more complex reasoning or inference, such as those requiring:
Question: Why was the shooting embarrassing for Xi Jinping?
Predicted Answer: No answer found.
True Answer: The shooting happened just before Xi Jinping's visit to the United States.
In this case, the system failed to connect the information it had about the time of the shooting to the visit of Xi Jinping. This is not directly relevant information in the article and is something that would have to be inferred from the context. The next potential problem would be the vocabulary overlap between the question and the part of the article from which the system retrieves its answer. This would be exacerbated if the question had a different form of words or different words altogether from those appearing in the crucial part of the relevant section of the article. This is likely with the way many of these questions are worded.
Several directions for future research and development could be:

- Continuing fine-tuning the BERT model on additional question-answering datasets that require reasoning about the answer of a more complex nature of questions, such as HotpotQA or Natural Questions. This may allow the model to handle more complex and open-ended questions.
- Experimenting with other model architectures and pre-training strategies—for instance, it is possible that a variant of BERT, RoBERTa or ALBERT, or a model designed explicitly for question answering, such as BART or T5, which performed very well in several language understanding benchmarks, will perform much better in the system in terms of accuracy and robustness.
- Applying techniques from information retrieval, such as document ranking and passage retrieval, to identify the most relevant articles and passages for a given question. This could prove especially useful when handling questions requiring synthesis from several parts of an article or from multiple articles.

In strong contrast to the developed systems, the current one barely performs fairly, although it will be a promising starting point for research and development into question-answering. Making proper use of nowadays applications with modern language models like BERT, and fine-tuning them on a corpus of free, unstructured text, allows one to build systems capable of extracting information out of it nowadays. Indeed, with the field of natural language processing rapidly developing, one can expect question-answering systems to become more and more advanced and flexible, able to deal with a wider range of questions and domains, from simple factoid questions up to quite complex open-ended queries implying reasoning and inference. Finally, we aim to develop question-answering systems that would truly understand and respond to natural language questions by providing accurate, informative, and context-relevant

answers that help users easily access knowledge within very extensive data in text form. Associated question-answering systems can be developed using the strength of pre-trained language models coupled with IR techniques, which are technology advanced yet also hands-on for the end user. Having said that, ladies and gentlemen, the tools and techniques that will be used in the current project make worth mentioning the following choice: the choice of using BERT in this project as the underlying language model. BERT was able to gain much popularity within the natural language processing community by proving its strong points on different tasks, among them answering questions. The bidirectional nature of the model is also helpful for understanding the subtlety of a natural language query to determine the relevance of a piece of information in a broader context. On the other hand, BERT is not always the only way to encode text to perform a task of question-answering. There is also a wide family of commonly used approaches known as baselines that utilize sentence embeddings. Sentence BERT refers to the fixed-size vector representation from a sentence or paragraph that can entail the meaning of that sentence or paragraph. For this task, two models have been proposed: SBERT and USE, with excellent performance on semantic similarity and information retrieval tasks.

However, in this work, relevant use of sentence embeddings in carrying out the incoherency between passages would certainly benefit the system to help ferret out the more relevant passages. That is to say, for those kinds of questions where the relations among sentences in a paragraph are important, the sentence or paragraph representations are optionally compared to the question embedding in a way that helps the system retrieve and rank these pertinent parts of the article. One of the positive sides of using models for sentence embedding, like SBERT, is that the methods are specifically developed to be computationally efficient and take care of large-scale datasets with minimal memory and processing needs. The reason for this is that in this case, the embeddings of sentences are precomputed and stored, so the retrieval and comparison of sentences are quite fast at inference time. On the other hand, BERT would need to process every single article through the model for every new query to do passage retrieval on that article—computationally expensive and time-consuming. However, one has to consider that choosing the text representation based on the task requirements and, in general, the characteristics of the dataset is among the most important. Sentence embeddings can be realized to be more suitable for passage retrieval, whereas BERT's usage of fine-grained attention and context over words can be more suitable for the actual question-answering task, which might be more beneficial, as an important aspect of the question-answering task is understanding the relationships between words and phrases. In this way, the usage of BERT for both passage ranking and answer extraction in this project provides a unified approach to leverage the model's capabilities in natural language understanding. In any case, this will be very important, since effective usage of sentence embeddings for effective passage retrieval will find great usage in the future to scale systems for large datasets and complex questions.

4. Conclusion

This work demonstrates the development and testing of a question-answering system for news articles using the BERT-Large Pre-trained Language Model. The system can extract relevant heads and answers, fine-tuned by BERT on the SQuAD dataset and appropriate to respond to natural language questions, quite accurately. The following are the main contributions and prime findings:

- This project implements a question-answering pipeline, capturing state-of-the-art NLP techniques through the BERT model, its release in particular, in an attempt to understand and answer questions based on a large corpus of unstructured text data.
- We evaluate the exact match accuracy of the system. It is the most stringent measure for any question-answering system to provide correct and complete answers. Indeed, 15% can be considered a bad result, indicating that this approach has great room for improvement, but it also means that this way of modeling is really showing its potential.
- Basically, qualitative analysis of the system predicted answers is important for establishing current state and road aheads, pointing out strong and weak points of the current approach, and indicating that more factors than simple exact match accuracy have to be considered.
- Finally, some promising research avenues and venues for development are fine-tuning on additional question-answering datasets, on the one hand, and result in alternative model architectures and pre-training strategies used in a better way to information retrieval and knowledge representation, leading to more sophisticated answer-generation and ranking-oriented methods, among others.
- It discusses the selection of BERT as the underlying language model and the advantages and trade-offs of using sentence embeddings like SBERT to allow efficient passage retrieval in question-answering systems.

This project broadly relates to the domain of Natural Language Processing and Information Retrieval and shows the potential for advanced language models like BERT on question-answering tasks, providing insights and future research directions in this area. As the scale of unstructured text data will keep increasing in size, the ability of effective and accurate extraction of relevant information residing in them, through natural language queries, will progressively become more and more important.

This research on question-answering systems, coupled with the understanding and interaction with human language, holds much promise in revolutionizing the means through which information is drawn, where the response needed could be found quickly and easily from extremely vast knowledge repositories. Bettering the design of question-answering systems with technical sophistication and practical utility will require a cross-disciplinary research approach in the disciplines of computer science—with the major force in this area—linguistics, information science, and domain expertise from the relevant fields.

References

[1] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers) (pp. 4171-4186).
[2] Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (pp. 2383-2392).
[3] Reimers, N., & Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 3973-3983).
[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).
[5] Yang, W., Xie, Y., Lin, A., Li, X., Tan, L., Xiong, K., ... & Lin, J. (2019). End-to-end open-domain question answering with BERTserini. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations) (pp. 72-77).