University *of Ljubljana*
Faculty *of Computer and Information Science*

Cross-lingual Question Answering

Nina Hostnik, Ajda Markič, Benjamin Plut

Abstract

tbd...

Keywords
Keyword1, Keyword2, Keyword3 ...

Advisors: Slavko Žitnik

Introduction

With the amount of websites and data sources on the internet rapidly increasing with each passing year, it's becoming harder and harder for users to find answers to any questions they might have [1]. They want short and precise answers instead of having to sift through long documents and extraneous information, as this process can be quite time consuming due to the colossal amount of data on the internet. This is why Question Answering Systems (QAS from here on) are becoming more and more necessary. QAS enable the user to input questions in natural language. They then compute them, find the answer in their databases or on the internet and present it back to the user in the form of a natural language answer. In order to complete this task, QAS must be able to understand natural language. This can prove to be a challenge [2], as computers cannot yet process natural language as easily or effectively as humans. And while progress is being made, it is often limited to the English language. Most datasets used for training language models used to train QAS are made for the English language and while there exist some datasets for other languages, they often contain far less documents and may not offer sufficient training to language representation models.

This lack of datasets in other languages is often solved by translating documents from English into the target language. However, the results obtained often depend on the quality of translation. In this project, we train a BERT language representation model [3] on a fully English dataset, a machine translated fully Slovene dataset, and a mix of both. We then evaluate the quality of the question answering task performed on a Slovene language corpora.

Related works

Typically, cross-lingual QAS will solve the issue of having two languages by first translating one into the other using machine translation [4]. However, the success of such an approach is limited by the quality of the translation itself. Another possible approach is to consider alternate translations of target words. However, the alignment information for extracting target-source word pairs necessary in training such models makes this approach rather difficult. Deep neural networks have been employed to better the results of information retrieval(IR), using either pre-trained word embeddings or training based on IR objectives. While the latter have shown impressive results on monolingual datasets, their reliance on large amounts of annotated data makes it unclear whether they can perform as well on often much smaller cross-lingual datasets.

Recently however, pre-trained language models such as BERT [3] have been developed, that have outperformed traditional word embeddings. BERT models the underlying data distribution in languages to its linguistic patterns. It has successfully been applied to monolingual IR and was extended to perform ranking in cross-lingual IR, without the need for alingment data. It has also been used in the development of an evaluation dataset for multilingual QAS.

We found three appropriate monolingual question answering datasets and one multilingual. The most important requirement is that answers to the questions aren't scattered across multiple paragraphs. We allowed datasets where answers were sometimes scattered over multiple sentences. The first of the datasets we considered is SQuAD [5], a dataset consisting of over 100.000 question-answer pairs. Questions and answers were extracted from Wikipedia article paragraphs by crowdworkers. Each question was then given at least two additional answers, except in rare cases when the crowdworker could

not find an answer. In the first version of SQuAD, there are no unanswerable questions, but they were added in SQuAD 2.0. Questions were divided into sets by dividing original articles - 80% of Wikipedia articles form the training set, 10% development set, and 10% a test set.

Similar datasets are QuAC and CoQA [6], which we do not consider appropriate because of their structure - questions can co-reference previous ones and yes or no answers are possible.

Another possible dataset is Google's Natural Questions (NQ) [7]. The dataset consists of sets of questions, Wikipedia pages that supposedly contain the answer, long answers, and short answers to the question (one of each per set). The source for the questions are google queries (8 words or longer and a specific set of rules to determine if the query is a question) that had a Wikipedia page in the first 5 search results. The long and short answers are extracted from said Wikipedia page and can be NULL if the answer is not found on the page. The dataset consists of 307.373 training examples with single annotations, 7.830 examples with 5-way annotations for development data, and 7.842 examples with 5-way annotations as test data.

Next possible dataset is TriviaQA [8]. Questions and answers were extracted from 14 different trivia and quiz websites and were then given a single (combined) evidence document from Bing search results or multiple evidence documents by using Wikipedia alone. Together there are 95.000 question-answer pairs that are organized into 650.000 training examples containing one document and 78.000 examples containing multiple Wikipedia articles. The problem with this dataset is that most of the automatically gathered evidence documents are large-scale and noisy, except 1975 human-annotated question-document-answer triples. Nearly 25% of the questions do not contain a clear answer in the evidence document.

Lastly, there is the multilingual dataset - Multilingual Knowledge Questions and Answers (MKQA) [9]. The dataset consists of 10.000 NQ questions that were translated into 26 languages or dialects from 14 language family branches. Answers are marked as either atomic value, entities, yes/no, short or long answers, or unanswerable. However, the dataset only contains questions and answers and would require the NQ dataset to provide Wikipedia articles.

Methods

0.1 Data

We decided to use Squad v2 for our dataset due to problems with Google's NQ. We have, however, removed all of the unanswerable questions. We extracted data from JSON and translated it into Slovene using the CEF eTranslator [10]. We formatted both the original and the translated version into a new JSON format, seen in figure 1, to turn the many-layered data into one layer that contained context, question, answers and title for each separate question, as shown in figure 2.

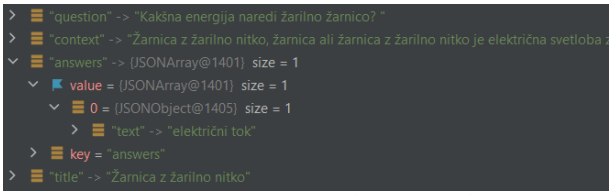

Figure 1. JSON data example

```
DatasetDict({
    train: Dataset({
        features: ['question', 'context', 'answers', 'title'],
        num_rows: 129670
    })
    validation: Dataset({
        features: ['question', 'context', 'answers', 'title'],
        num_rows: 11820
    })
})
```

Figure 2. Shape of the Dataset type after loading the JSON files.

0.2 Model

We used a pretrained BERT for QA model from TensorFlow [11], using a pre-trained checkpoint of a cased multilingual BERT model from Hugging Face [12] and fine-tuned it on our data using the Transformers [13] library.

0.3 Preprocessing

The data was preprocessed and turned into an input that the QA model could use. The inputs consisted of a tokenizer class containing the questions and context data, as well as the maximum allowed length of the input, a turncation method, offset mapping and padding method. Next, the answers were mapped to the context and the start and end position was added to input. Answers were mapped to the context by finding the start and end tokens of the answer in the context. Despite all the questions in the dataset having an answer, this may no longer be the case after preprocessing. The model has a fixed maximum input it can handle and the context may sometimes be too long. By setting *turncation* to 'only-second' that only some of the context is lost while the question remains. However, if the answer was contained in the turncated part of the context, this effectively made the question unanswerable. Because those answers could no longer be mapped to context, they were labeled (0, 0). The computed start and end positions were then added to the input.

0.4 Training

Model training was done on a PC with an AMD Ryzen 7 3700x processor, 16GB of RAM and a Nvidia RTX 3080 10GB graphics card. We used a batch size of 8 because of GPU VRAM limitations. For now we decided to use the number of epochs as 3, but we might change that later depending on results. For initial testing purposes and time constraints we only used 1 epoch which took around 1 hour and 15 minutes to complete. This scales linearly with the increase in the number of epochs.

0.5 Comparing models

For the comparison of models we decided to use the F1 score and accuracy. We will be testing each of our finished models on a Slovene and English validation set and comparing the F1 scores and accuracy results for each.

Discussion

tbd...

Acknowledgments

tbd...

References

[1] Poonam Gupta et al. A survey of Text Question Answering Techniques. *International Journal of Computer Applications*, 2012.

[2] Patrick Lewis et al. MLQA: Evaluating Cross-lingual Extractive Question Answering. *arXiv*, 2020.

[3] Jacob Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv*, 2019.

[4] Zhuolin Jian et al. Cross-lingual Information Retrieval with BERT. *arXiv*, 2020.

[5] Pranav Rajpurkar et al. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

[6] Mark Yatskar. A Qualitative Comparison of CoQA, SQuAD 2.0 and QuAC. *arXiv*, 2019.

[7] Tom Kwiatkowski et al. Natural Questions: a Benchmark for Question Answering Research. *Transactions of the Association of Computational Linguistics*, 2019.

[8] Mandar Joshi et al. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *arXiv*, 2017.

[9] Shayne Longpre et al. MKQA: A Linguistically Diverse Benchmark for Multilingual Open Domain Question Answering. *arXiv*, 2021.

[10] CEF eTranslator. *Dostopano: 28. 4. 2022*.

[11] TensorFlow. *Dostopano: 3. 5. 2022*.

[12] Hugging Face. *Dostopano: 3. 5. 2022*.

[13] Transformers. *Dostopano: 3. 5. 2022*.