Dialectal Extractive Question Answering (DialQA)

Checkpoint 2: SOTA Baseline Implementation

Adel Alkhamisy, Aniket Pandey, Pankaj Jatav, Yash Bagal

Introduction

Our goal in doing this work is to create a Dialectal Extractive Question Answering system that can recognize different languages dialects and output an answer in response to a question. We used the baseline Dialectal Extractive Question Answering (DialQA) in this experiment, which was suggested by (Faisal et al., 2021). Moreover, we have used SD-QA[1] dataset and the Google's open source BERT model of pre-training language representations to implement the baseline. We obtained approximately 70% F1-Score in all languages which are considered accurate compared to the baseline model that fluctuates between 53.4% - 69.2% in all languages. Unlike DialQA implemented by (Faisal et al., 2021) which uses a minimal BERT model, we plan to implement a BERT with Language-Clustered vocabulary (Chung et al., 2020) or Poolingformer (Zhang et al., 2021) in an effort to improve the performance.

Approach

Our approach to implement the baseline model was simple, first we obtained the gold dataset from SD-QA[1]. The dataset has development and test data for 5 varieties of English (Nigeria, USA, South India, Australia, Philippines), 4 varieties of Arabic (Algeria, Egypt, Jordan, Tunisia), and 2 varieties of Kiswahili (Kenya, Tanzania). In our next step to implement the baseline, we first used the BERT method of pre-training language representations which obtains state-of-the-art results on a wide array of Natural Language Processing tasks.

The BERT model open source was obtained from the Google Research GitHub, BERT repository[2]. It is a method of pre-training language representations which obtains state-of-the-art-results on a wide array of Natural Language Processing tasks.

Once our complete setup was done, we used the gold passage baseline by google-research-datasets/tydiqa[3] with some minor modification to run the BERT model. We used gold passage task because it is simplified and only the gold answer passage is provided.

To run and evaluate the baseline implementation please check the README.md file inside the code folder with all the steps mentioned there. For now we are submitting zip file because of issues while fixing code however we plan to create a git repo and push our work on GitHub for final checkpoint 3 step.

Experiment and Analysis

In the starting we were facing difficulty using the entire SD-QA dataset which included varieties of languages as specified in our project proposal. So, we then switched to gold data which we obtained from DialQA.

We used the BERT model as it was described in paper from google-research/bert. From our previous learning in HW2: Implementing a minimal BERT we found that in optimization step changing the Beta 1 and Beta 2 value has considerable effect on the accuracy. So, we decided to try this on Project and the result is still inconclusive which we are planning to continue working on it.

While working with BERT we also found that it uses TensorFlow 1.15 which caused us many version conflicts. Similarly, we had issues with TensorFlow Protobuf package.

Performance

Our work produced close to the baseline accuracy. On all Languages average 70% F1 Score. We are still tunning the hyper-parameters and exploring the way to find Exact Match and Example Count as it is mentioned in baseline results.

Further Work for Checkpoint 3

Through our experiments and research, we have identified several ways to improve the accuracy which are listed below:

- As proposed in proposal we will try BERT with Language-Clustered vocabulary as proposed in (Chung et al., 2020)[4].
- As proposed in proposal we will also work with Poolingformer (Zhang et al., 2021)[5], Long Document Modeling with Pooling Attention.
- The gold data in DialQA work by ffaisal93/DialQA consist of all languages in one JSON file, we are planning to segregate and test the data on individual language.
- In our NLP HW1 we saw that using the pretrained GloVe improved the score significantly and therefore following the same path we will use google/bert_uncased_L-12_H-768_A-12[6] pretrained model for training and see if the result improves. The model is a set of 24 BERT models referenced in Well-Read Students Learn Better: On the Importance of Pre-training Compact Models. They are most effective in the context of knowledge distillation, where the fine-tuning labels are produced by a larger and more accurate teacher.
- Lastly, depending upon the result and work we will experiment with hyper-tunning the parameters.

Contribution

We adopted the mechanism of pair programming where we regularly collaborated with each other and figured out a way to overcome the issues. We worked together on almost all aspect of research and implementation together and a broad level overview of work done until checkpoint 2 and work to be done next is listed below.

Team Member	Checkpoint 2	Final Checkpoint 3
Adel Alkhamisy	Worked on texting, evaluating, and generating performance matrix.	Segregating the data and figuring out the approach for hyper-tunning the parameters.
Aniket Pandey	Implementing BERT, preprocessing and understanding the dataset. Creating a runnable code for baseline implementation.	Checking the feasibility of using the BERT with Language-Clustered vocabulary and implementing the same.
Pankaj Jatav		Implementing the Poolingformer as proposed in Zhang et al and figuring out a way to solve the existing issues involved.
Yash Bagal	Preprocessing the dataset and research for checkpoint 3.	Work on using the Google BERT Uncased pretrained model.

References

1. https://github.com/ffaisal93/sd-qa
2. https://github.com/google-research/bert
3. https://github.com/google-research-datasets/tydiqa/tree/master/gold_passage_baseline
4. Hyung Won Chung, Dan Garrette, Kiat Chuan Tan, and Jason Riesa. 2020. Improving multilingual models with language-clustered vocabularies.
5. Hang Zhang, Yeyun Gong, Yelong Shen, Weisheng Li, Jiancheng Lv, Nan Duan, and Weizhu Chen. 2021. Poolingformer: Long document modeling with pooling attention.
6. https://huggingface.co/google/bert_uncased_L-12_H-768_A-12