EE 599 Deep Learning – Revised Project Proposal

Wenjing Lin, Jiaqi Liu

April 18th, 2020

Project Title: Named Entity Recognition with pretrained BERT

Project Team: Wenjing Lin (wenjingl@usc.edu), Jiaqi Liu (jliu9289@usc.edu)

Project Summary: In this project, we will summarize several NLP techniques, including statistic language representation methods like Word2vec[1] (CBOW and skip grams), GLOVE[2], the contextualized word embedding like CoVe[3] and ELMO, Bidirectional Encoder model BERT[4] and transformer. Then we will choose one of them as base model, transfer it to achieve Named Entity Recognition (NER) task in text file dataset.

The traditional NLP models use statistic word embedding method (Word2vec) for word vector initialization. Through pretraining, the related word vectors are represented close to each other. But in this method only output word vectors are useful. So, for different corpus, it needs to train from scratch which cost much time and computation. Also, words are not isolated but are always context-dependent. The Word2vec method fails to capture word meaning in different contexts. The later NLP models are contextualized and don't need to always train from scratch, like Cove model use fine-tune strategy, ELMO model use transformer. But they are all unidirectional or shallowly bidirectional. In these models, the later token can only be predicted by the previous tokens. This will restrict a token-level comprehension, where the context from both directions is crucial.

BERT model is a technique to apply deep bidirectional pretrain from unlabeled text developed by Google. BERT is unsupervised. It can learn from unlabeled text, which save much expense. The key innovation in BERT is that it trains text sequence with both left to right and right to left directions. It outperforms previous methods in QA task and task for extracting information. What's more, BERT is based on transformer and attention mechanism, so the pre-trained BERT model works good for downstream NLP tasks.

NER, also known as entity identification, is probably the very first step during natural language processing process. It can be used to extract information like name of persons, companies and organizations. In our project we choose pretrained BERT as our base model for named entity recognition task.

Our goal is to understand the theoretical claims in above papers and clearly explain NLP methods including Word2vec, GLOVE, ELMO, transformer and BERT. We will also

implement BERT in NER task. A successful result is given an unstructured and unannotated text file, the model can output an annotated text file, in which all name (e.g. persons, locations) are marked with pre-designed categories.

Data Needs and Acquisition Plan: Enron Emails is a dataset includes over 500,000 email messages tagged with names, dates and times collected and prepared by the CALO Project. It contains data from about 150 users, mostly senior management of Enron, organized into folders. The corpus contains a total of about 0.5M messages.
Dataset website: https://www.kaggle.com/abhinavwalia95/entity-annotated-corpus/a

Primary References and Codebase: We propose to build on the approach used in:

- [1] Mikolov, T. et al. (2013) Efficient Estimation of Word Representations in Vector Space. arXiv.org. [online]. Available from: http://search.proquest.com/docview/2086087644/.
- [2] Pennington, J., Socher, R. and Manning, C.D., 2014, October. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).
- [3] McCann, B., Bradbury, J., Xiong, C. and Socher, R., 2017. Learned in translation: Contextualized word vectors. In Advances in Neural Information Processing Systems (pp. 6294-6305).
- [4] Devlin, J. et al. (2018) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.
- Blog Post, Veysel Kocaman, "Named Entity Recognition (NER) with BERT in Spark NLP"
- GitHub code base: TensorFlow code and pre-trained models for BERT, google-research/Bert, NER with BERT in Spark NLP

Architecture Investigation Plan: We will firstly review and learn some key concepts including LSTM, attention mechanism, transformer. We will also dig into these NLP models we mentioned above and do some summary for the theoretical report.

For the pretrained BERT application, the project will base on pretrained BERT model in TensorFlow Hub and will probably use Spark to mark entity name. We will build up a baseline model with pre-trained BERT to do Named Entity Recognition job, perform experiments on various parameters and optimize the prediction accuracy and generation. We will Finetune BERT as following steps:

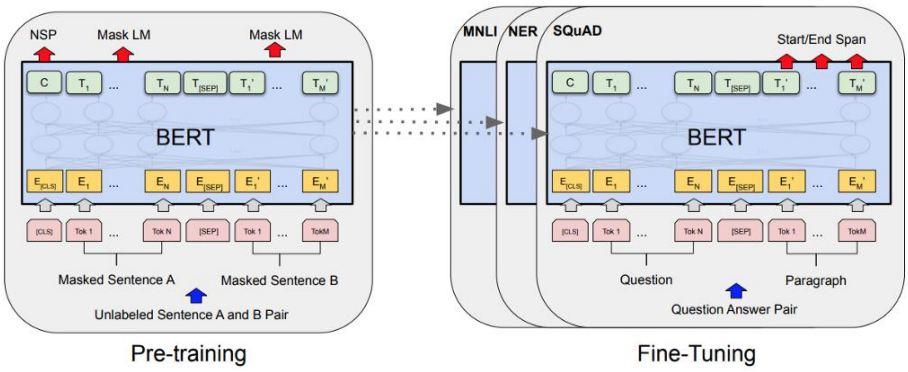

Figure 1: Overall pre-training and fine-tuning procedures for BERT [4]

1. Preprocess Enron Emails dataset for training.
2. Initialize pre-trained model and set the hyperparameters.
3. Convert Enron Emails dataset to tensors and load tensors.
4. Train and validate over several epochs, monitor performance at each epoch (e.g. loss, precision/recall, confusion matrix).

Example of result:

Input: Peter park is a nice man lives in New York.

Output: Exact person and location name like:

```
+------------+------+
|chunk       |entity|
+------------+------+
|Peter Parker|PER   |
|New York    |LOC   |
+------------+------+
```

Figure 2: example output

If time permits, we would also like to perform streaming process.

Milestones:

- By 04/23, achieve a bugless NLP training model with BERT (regardless name entity and accuracy)
- By 04/22, draw up report on Word2vec and GLOVE
- By 04/27, apply the model to our NER task
- By 04/28, draw up report on CoVe and ELMO
- By 04/30, try different parameters and optimize model, improve accuracy
- By 05/01, draw up report on transformer and BERT
- By 05/05, prepare for presentation

- If have time, we will try stream process model
- Before 05/11, prepare for project report
- Before 05/13, prepare for project video

Estimated Compute Needs: Given that the whole dataset is about 1.7 G, we plan to use Amazon EC2 P3 Instances. Use spot price, the EC2 p3.2xlarge is about $1 per hour. The total credits of our team allow training for 90 hours. We also set up a shared notebook in Google Colab.

Requested Mentor with Rationale: We believe Oliver would be an appropriate team mentor due to his great work on signal processing and bidirectional neural networks algorithm fields.