Final Project Proposal for EECS 595

Jiachen Jiang
jiachenj@umich.edu

Chengtian Zhang
zctchn@umich.edu

1 Problem Statement

We plan to work on SemEval-2021 shared task 4: Reading Comprehension of Abstract Meaning (ReCAM)(Zheng et al., 2021).

Machine reading comprehension (MRC) tasks is designed to help evaluate the ability of machines in representing and understanding human languages and reasoning. Given a passage, the machine is expected to give the answers of questions related with this passage. Specifically, SemEval-2021 shared task 4 require the participating system to fill in the the correct answer from five candidates of abstract concepts in a cloze-style to replace the *@Placeholder* in the question. Instead of predicting concrete concepts in the previous work, in this task we ask models to choose abstract words removed from human-written summaries. There are there subtasks to evaluate the performance of the model based on two aspects of *abstractness*, *imperceptibility* and *nonspecificity*.

- **Subtask 1** focus on evaluating the system's ability in understanding *imperceptibility*, which are words that cannot be directly perceived in the physical world, e.g. service/economy compared with trees/red;.
- **Subtask 2** aims to measuring the system's ability in comprehending *nonspecificity*, which are nonspecific concepts located high in a hypernym hierarchy given the context of a passage, e.g. vertebrate compared with monkey.
- **Subtask 3** focus on the model's transferability over the two types of *abstractness*, we need to train the model on the Subtask 1 evaluate it on Subtask 2 and vice versa.

2 Proposed Approaches

We propose an approach based on pre-trained language models(LMs) and DUal Multi-head Co-Attention (DUMA) multi-choice classifier with negative data augment to get the final result. The overall architecture of our system mainly consists of three parts: Negative Augmentation, Pre-trained LMs and DUMA multi-choice classifier.

- **Negative Augmentation** According to Chen et al., 2020, stronger negative samples will help the model learning with better performance. So we plan to generate some negative words using the pre-trained LMs to help train the models. Specifically, we replace the *@placeholder* with [MASK] to reconstruct the input and ask the BERT model to predict the word token at the [MASK]. These generated words are used as negative candidates.
- **Pre-trained LMs** After negative augmentation, the sequence is fed through a Transformer-based encoder. In our case, we would try different encoders, such as such as BERT/ ALBERT/ ROBERT/ DEBERT/ ELECTRA and compare their performance. We would choose the pre-trained LM with best result. Note all pre-trained LMs have finished the task-adaptive training process.
- **DUMA Multi-choice Classifier** In this part, we would add an extra attention layer which is Dual Multi-head Co-Attention (DUMA) module as described in (Zhu et al., 2020). Basically, it involves 1) splitting the output sequence from the encoder into question-answer sequence and passage sequence; 2) calculating two attention representations from the two sequences, one from the passage attending the question-answer, the other vice versa; 3) concatenate the two attention representations together after individually mean-pooled; 4) The representations would be sent to the classifier. The answer option with the highest probability is picked as the predicted answer. We would

use the Cross Entropy function between the ground truth and the predicted probabilities to compute the loss.

3 Dataset Overview

- **ReCAM**. Dataset for the SemEval-2021 Task 4. Data is stored one-question-per-line in json format, including article, question, options and label. In Subtask 1, the training/ trail/ development/ test contains 3,227/1,000/837/2,025 instances. In Subtask 2, the training/ trail/ development/ test contains 3,318/1,000/851/2,017 instances.

- **CNN/Daily Mail**. It consists 300k unique news articles as written by journalists at CNN and the Daily Mail. We would use it to implement task-adaptive pretraining.

4 Previous Work

According to the type of the answer, Machine reading comprehension (MRC) tasks can be divided into the following three categories Chen, 2018.

- **Span prediction**. Extractive question answering requires the system to extract a suitable range of text fragments from a given original text based on the question as to the answer.(Hermann et al., 2015; Hill et al., 2016; Onishi et al., 2016; Rajpurkar et al., 2016; Trischler et al., 2017)

- **Free-form answer**: This task require models to generate an answer. It requires the system to mine deep-level contextual semantic information according to a given question to give the best answer.(Nguyen et al., 2016; He et al., 2018;Kociský et al., 2017)

- **Cloze-style**: The system must choose a word or entity from the set of candidate answers to fill in the ”@placeholder” in the question to make the sentence complete.(Hermann et al., 2015; Hill et al., 2016; Onishi et al., 2016;)

This task is similar to the Cloze-style task. Unlike previous work, ReCAM questions specifically focus on abstract words.

5 Implementation Plan

- **Data Pre-processing** For each passage, there is a summary with five candidate answers. We substitute the options into the summary to form several complete sentences. Then we concatenate the option-filled sentence and passage tokens as input samples, wrapped by [CLS] token and [SEP] tokens.

- **Task-adaptive Pretraining** We need to apply task-adaptive pretraining on LMs, to get better embedding Gururangan et al., 2020. Most LMs are trained in the general domain corpus such as Wikipedia. Task adaptive pre-training would use domain-specific data, e.g. CNN daily, to make the model better fit the distribution in the domain.

- **Fine-tuning** We would fine-tune our PLM encoder on ReCAM. We plan to implement with PyTorch to get the pre-trained language models, then use AdamW optimizer to fine-tune the models. We pick the best learning rate from the dev set and set batch size small due to the limit of GPU memory of Google Colab.

Here are the milestones for the rest of the semester:

- November 7: Get familiar with task-adaptive models, such as BERT, ALBERT, ROBERT etc.

- November 10: Finish the data preprocessing and negative data augment.

- November 17: Implement task-adaptive pre-train of BERT and ELECTRA on CNN daily.

- November 24: Construct the system and do fine-tune training using ReCAM.

- December 1: Get final Accuracy result and compare it with other papers.

- December 10 : Prepare for the presentation and final report.

6 Work Division

The are two people in our group and the work division are as follows:

- **Jiachen Jiang**. Implementation of task-adaptive pre-train, system construction.

- **Chengtian Zhang**. Data preprocessing, fine-tune training.

We would work together on presentation and final report.

References

Danqi Chen. 2018. *Neural Reading Comprehension and Beyond*. Ph.D. thesis. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2021-09-28.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, Xuan Liu, Tian Wu, and Haifeng Wang. 2018. DuReader: a Chinese machine reading comprehension dataset from real-world applications. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 37–46, Melbourne, Australia. Association for Computational Linguistics.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend.

Nathan R. Hill, Samuel T. Fatoba, Jason L. Oke, Jennifer A. Hirst, Christopher A. O'Callaghan, Daniel S. Lasserson, and F. D. Richard Hobbs. 2016. Global prevalence of chronic kidney disease – a systematic review and meta-analysis. *PLOS ONE*, 11(7):1–18.

Tomás Kociský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2017. The narrativeqa reading comprehension challenge. *CoRR*, abs/1712.07040.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268.

Takeshi Onishi, Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. 2016. Who did What: A Large-Scale Person-Centered Cloze Dataset. *arXiv e-prints*, page arXiv:1608.05457.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Boyuan Zheng, Xiaoyu Yang, Yu-Ping Ruan, Zhen-Hua Ling, Quan Liu, Si Wei, and Xiaodan Zhu. 2021. Semeval-2021 task 4: Reading comprehension of abstract meaning. *CoRR*, abs/2105.14879.

Pengfei Zhu, Hai Zhao, and Xiaoguang Li. 2020. Dual multi-head co-attention for multi-choice reading comprehension. *CoRR*, abs/2001.09415.