CS510 Project Proposal: Methods for Model Compression in Neural Information Retrieval

Daniel Campos
dcampos3@illinois.edu

ACM Reference Format:
Daniel Campos. 2021. CS510 Project Proposal: Methods for Model Compression in Neural Information Retrieval. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

In the past few years neural methods for NLP have shown tremendous progress as model size and training time grows. Models based our neural language models have proven tremendously successful for information retrieval but these large models can prove difficult to use due to large model footprint(> 400 mb commonly) and slow inference time. Seeking to remedy this researchers have employed knowledge distillation and layer dropping to make smaller models that approximate the performance of the larger models. In other research fields like computer vision methods like model pruning, which is the removal of network structures or neurons, has been used as an effective model compression mechanism. Thus, our research question is as follows:

How does model compression using layer dropping and network pruning effect the performance of Information Retrieval systems?

As the size of neural models grows past current 100B behemoths [2] inference becomes a bigger problem commonly requiring specialized hardware like clusters of GPUs and FPGAs. Recent work has shown that larger networks can learn quicker [9], are more accurate and are more sample efficient [6] which means scaling trends are likely to continue. Additionally, novel IR research is reaching industry at breakneck pace which means novel neural models are being served trillions of times a year to billions of users. Any method which allows for model compression and faster inference is immediate returns for any company deploying search and for the lower compute overhead on the world.

2 Prior Work

Compression of neural networks is by no means a new field of research and is extremely popular especially for fields like computer vision. Common methods for network compression are: quantization, distillation, layer dropping, and model pruning. In model distillation [1] the original network is used to train a smaller network to mimic the behavior of the large network. In model quantization [5] models are made smaller by reducing the numbers of bits that represent each weight. Model Pruning [8] has focused on finding sub-networks in the original network by structured and unstructured pruning. In structured pruning, successful sub-networks are found by removing neurons [11] or larger network specific structures like attention heads [10]. In unstructured pruning the successful sub-networks are found by setting individual weights to zero [7].

Currently, in IR research a the common approach for making models smaller is to drop layers from a neural model like BERT and then use model distillation with a large model (without dropping layers). This approach is effective as it is simple but the approach lacks granularity as most models only have 12 layers. If a model has a specific target size layer dropping only allows researchers to reach a rough approximation within 5-10%. Model pruning is well studied in computer vision but in the NLP domain and IR domain there has been little research. Thus, the novelty in our work is the lack of study and the lack of comparison with existing methods of compression.

3 Method

To explore how various compression methodologies behave we will train and evaluate various models on a well studied IR benchmark, the MSMARCO passage ranking dataset (MPR) [3]. The MPR is a dataset which originates from a question answering dataset. The original dataset consists of 8.8 million passages and 1 million user queries issued to a commercial search engine. For each query, a judge read the top 10 passages extracted by Bing and wrote an answer to the query and attributed the answer to one or more passages. The MPR is an adaptation of the question answering dataset to produce a passage ranking tasks. The document collection is a combination of all unique passages from the question answering task and the relevance judgements are binary labels representing which passage was used to generate the human generated answer. As the tasks is recall focused, the evaluation metric is Mean Reciprocal Rank (MRR) @1000.

Conference'17, July 2017, Washington, DC, USA
© 2021 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

The large data regime of this dataset has made it one of the most popular evaluation frameworks for neural IR with over 100 submissions and was used in the 2019, 2020, and 2021 TREC Deep Learning Track.

Since this benchmark is well established our work will focus on evaluating a broad and representative set of models. Our research will center around a second stage retrieval mechanism built on the BERT [4] architecture. To adapt the model for IR passage ranking is framed as a binary classification problem: give a query Q and a passage P predict a label for relevance. In practice, this mean concatenating each query and passage using a [SEP] token and passing this sequence through the language model. This method has lead to many state of the art results on the MPR and allows ample room for experimentation.

Once we have a robust baseline we will implement various methods of model compression. The first method, which is commonly used in industry, is layer dropping. We will produce models with 3, 6, 9, and 12 (original size) transformer layers and evaluate performance. The Second method, unstructured network pruning, will seek to both match the absolute size in parameters of the layer drop models. Using these 7 networks, we will present numerical analysis on the effectiveness of model compression.

4 Problem Description and Project Plan

In this project we seek to answer the following questions:

1. Can we achieve a high sparsity(80%+) on neural IR methods and retain performance?
2. How do results of IR pruning compare to computer vision and other NLP tasks.

4.1 Work Plan

Our work plan is broad but is laid out below. We may decrease the amount of pruned models and models implemented based on compute availability and coding efficiency. Since Daniel is the only team member he will do everything

1. Initial dataset exploration - March 30th
2. Baseline model running and evaluated - April 15th
3. Models pruned and evaluated - April 30th
4. Presentation of results and paper - May 5th

References

[1] Jimmy Ba and R. Caruana. 2014. Do Deep Nets Really Need to be Deep? *ArXiv* abs/1312.6184 (2014).

[2] T. Brown, B. Mann, Nick Ryder, Melanie Subbiah, J. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, G. Krüger, T. Henighan, R. Child, Aditya Ramesh, D. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, E. Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, J. Clark, Christopher Berner, Sam McCandlish, A. Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *ArXiv* abs/2005.14165 (2020).

[3] Daniel Fernando Campos, T. Nguyen, M. Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, L. Deng, and Bhaskar Mitra. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *ArXiv* abs/1611.09268 (2016).

[4] J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

[5] Song Han, Huizi Mao, and W. Dally. 2016. Deep Compression: Compressing Deep Neural Network with Pruning, Trained Quantization and Huffman Coding. *arXiv: Computer Vision and Pattern Recognition* (2016).

[6] J. Kaplan, Sam McCandlish, T. Henighan, T. Brown, Benjamin Chess, R. Child, Scott Gray, A. Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling Laws for Neural Language Models. *ArXiv* abs/2001.08361 (2020).

[7] S. Kwon, D. Lee, Byeongwook Kim, Parichay Kapoor, Baeseong Park, and Gu-Yeon Wei. 2019. Structured Compression by Unstructured Pruning for Sparse Quantized Neural Networks. *ArXiv* abs/1905.10138 (2019).

[8] Y. LeCun, J. Denker, and S. Solla. 1989. Optimal Brain Damage. In *NIPS*.

[9] Zhuohan Li, Eric Wallace, Sheng Shen, Kevin Lin, K. Keutzer, D. Klein, and J. Gonzalez. 2020. Train Large, Then Compress: Rethinking Model Size for Efficient Training and Inference of Transformers. *ArXiv* abs/2002.11794 (2020).

[10] Elena Voita, David Talbot, F. Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. In *ACL*.

[11] H. Wang, Qiming Zhang, Yuehai Wang, and H. Hu. 2019. Structured Pruning for Efficient ConvNets via Incremental Regularization. *2019 International Joint Conference on Neural Networks (IJCNN)* (2019), 1–8.