SparseBERT: Efficient Transformer-based Language Model with Integrated Pruning

Sakshi Bhavsar
New York University
New York, NY, USA
sgb9086@nyu.edu

Shruti Pangare
New York University
New York, NY, USA
stp8232@nyu.edu

***Abstract*—Transformer-based language models have achieved remarkable success across various natural language processing tasks. However, their substantial model sizes and computational demands pose challenges for deployment on resource-constrained devices. This project introduces SparseBERT, an efficient transformer-based language model that incorporates pruning techniques directly into the training process. Our objective is to significantly reduce model size and computational requirements while maintaining performance on downstream tasks, thereby enhancing the suitability of transformer models for deployment in resource-limited environments.**

I. Introduction

Transformer architectures, exemplified by models like BERT, have set new benchmarks in natural language processing (NLP). Despite their effectiveness, the extensive parameter counts and computational burdens associated with these models limit their applicability in scenarios with constrained resources. To address this, we propose SparseBERT, which integrates pruning mechanisms during training to achieve a more compact and efficient model without compromising performance.

II. Challenges

Developing SparseBERT entails addressing several key challenges:

- **Balancing Accuracy and Efficiency:** Identifying optimal pruning rates that minimize model size without significantly degrading performance.
- **Dynamic Pruning Integration:** Designing a training methodology that seamlessly incorporates pruning mechanisms into the learning process.
- **Architecture Sensitivity:** Determining which components of transformer models (e.g., attention heads, feed-forward network (FFN) layers) are most amenable to pruning.
- **Reactivation Mechanism:** Implementing strategies to reactivate previously pruned connections when they become important.
- **Evaluation Complexity:** Developing comprehensive benchmarks that assess both model performance and efficiency metrics.

III. Proposed Approach

To tackle these challenges, we propose the following approaches:

A. Importance Scoring Mechanisms

We will implement multiple criteria to evaluate the importance of attention heads and FFN neurons during training, including:

- Gradient-based importance scoring
- Activation-based relevance assessment

B. Progressive Pruning Schedule

We will design a curriculum that gradually increases sparsity throughout training, following:

- Cubic sparsity progression
- Sigmoid-based pruning growth

C. Structured Pruning

To maintain hardware efficiency, we will focus on structured pruning techniques, such as:

- Removing entire attention heads
- Reducing FFN dimensions

D. Dynamic Masking and Reactivation

We will create mechanisms to:

- Mask less important weights during training
- Allow important connections to be reactivated

E. Loss Function Regularization

We will augment the training loss with sparsity-promoting regularizers to encourage model compression, including:

- L_1 and L_0 norm-based regularization
- Group sparsity constraints

IV. Hardware Framework

- **Compute**: NVIDIA GPUs (Tesla V100, A100) for parallel processing.
- **Cloud, Edge based** : NYU HPC

V. Software Framework

Our implementation will utilize the following software components:

- **Framework:** PyTorch with the Hugging Face Transformers library
- **Existing Code to Reuse:**
  - Transformer implementations from Hugging Face
  - PyTorch's pruning utilities
  - Custom training loops with pruning logic

We will evaluate SparseBERT using the GLUE benchmark, focusing on tasks such as:

- Multi-Genre Natural Language Inference (MNLI)
- Quora Question Pairs (QQP)
- Stanford Sentiment Treebank (SST-2)

VI. Demo Planned

Our demonstration will showcase:

- **Comparison with BERT-base:** Highlighting parameter count reduction, inference speed improvements, and memory footprint reduction.
- **Visual Analysis:** Illustrating which model components were pruned and their effect on performance.
- **Interactive Exploration:** Allowing exploration of the accuracy-efficiency tradeoff with different pruning configurations.
- **Live Inference:** Demonstrating real-time inference on resource-constrained hardware.

References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, MN, USA, 2019, pp. 4171–4186.

[2] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both Weights and Connections for Efficient Neural Networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 1135–1143.

[3] V. Sanh, T. Wolf, and A. M. Rush, *"Movement Pruning: Adaptive Sparsity by Fine-Tuning," in Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020.

[4] Mitchell A. Gordon Kevin Duh Nicholas Andrews *Compressing BERT: Studying the Effects of Weight Pruning on Transfer Learning*

[5] Victor Sanh1, Thomas Wolf1, Alexander M. Rush1,2 *Movement Pruning: Adaptive Sparsity by Fine-Tuning*