Final Year Project – Batch 1

Thresholding Binarized Neural Network to Improve Accuracy in LLM training

Team Members

1. 106120014 - Arpon Kapuria
2. 106120025 - Brintha M
3. 106120135 - Udipta Pathak

Project Guide

Prof. N Ramasubramanian

Department of Computer Science & Engineering

National Institute of Technology, Tiruchirappalli - 620015

Introduction

Large Language Models are powerful models that are trained on large amount of data to perform human level task.

Ex: ChatGPT, Gemini AI

Transformers work as their backbone architecture. Transformers have become SOTA.

However their large model size and even larger runtime usages causes trouble in training and inference, which is why optimization is important.

Different techniques are used like mixed precision training, knowledge distillation, pruning quantization during training to reduce memory requirements, model size and increase speed on hardware for computations.

Literature Survey

Paper Title	Methodology	Drawbacks
Train Large, Then Compress: Rethinking Model Size for Efficient Training and Inference of Transformers, **[3]**	Studies the impact of model size reduction. Explains why faster convergence during training is helpful. Explains the proper setting for epochs.	Only explains it through pruning and quantization. Observation to different compression methods or model architectures needs further investigation.
Efficient Transformer Knowledge Distillation: A Performance Review **[4]**	Provides a detailed process of knowledge distillation on transformers. Proposes a new dataset explaining different performances and tradeoffs.	Performance degradation is noticed in one of the methods. No explanation is provided, which requires further investigation.
Understanding INT4 Quantization for Transformer Models: Latency Speedup, Composability, and Failure Cases **[5]**	Explores the feasibility of INT4 quantization on weight and activations	Optimization method may not be suitable for all transformer models like decoder only, due to accuracy degradation.
Binarized Neural Networks: Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1 **[6]**	Explores Binarization on neural network models,	Although it provides a better reduction on model size but we can notice huge accuracy degradation.
Understanding the difficulty of training deep feedforward neural networks. **[7]**	Proposes a new initialization technique “Xavier Initialization” for neural network training which keeps the variance of the outputs from each layer roughly equal.	Although it provides better results then other methods, it is not good for ReLU because it does not account for the non-zero mean output and sparsity of ReLU activations.

Research Gaps

As we explore different optimization techniques,

We find that Quantization is quite effective and sometimes matches the performance of the baseline model.

Moving on to Binarization takes us a step further in optimization. Binarization offers even better compression than quantization, but it comes with a drawback of accuracy loss.

Our goal is to address this challenge and finds ways to maintain accuracy while benefiting from the compression advantages of Binarization.

Problem Statement and Objectives

As we mentioned the gap and disadvantages in previous pages,

Therefore it is important to compress the model to lower it’s hardware cost and accelerate the model training for faster learning.

We aim to solve the issue by compacting the data to train and infer on transformer using techniques like quantization and binarization.

First we explore different quantization techniques then we apply them on transformers to check sensitivity of different parts of transformer.

Finally we present a novel technique that is a compromise between quantization and binarization, achieving the best of both.

Proposed Methodology

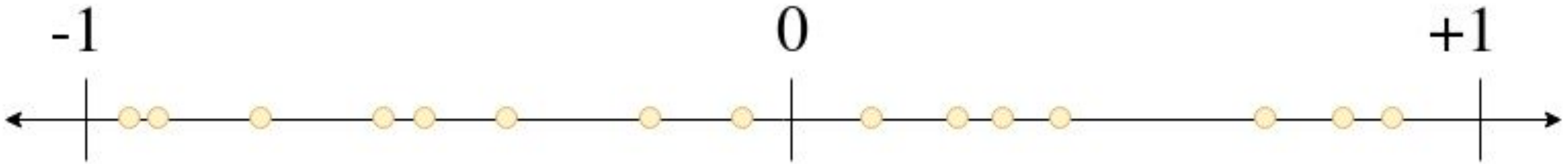
We are trying to achieve a trade off between quantization which compresses data to some extent, and binarization that compress data to extreme.

We propose a method that is more compressed than quantization and more accurate than binarized method.

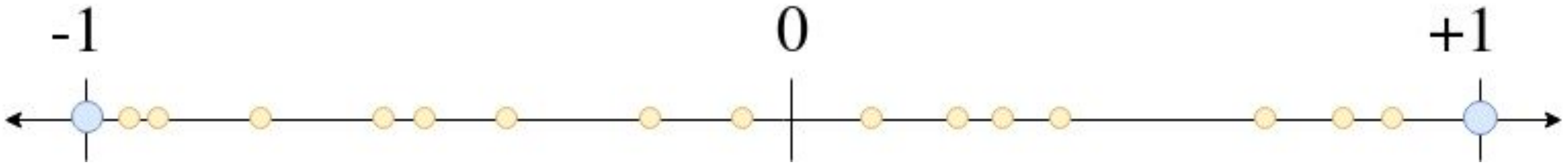
"Instead of mapping weights to -1 and 1 like we do in binarized neural network, we use mean of weights from xavier initialisation to get more accurate representation of weights without losing much information."

Proposed Methodology

Normalized Data

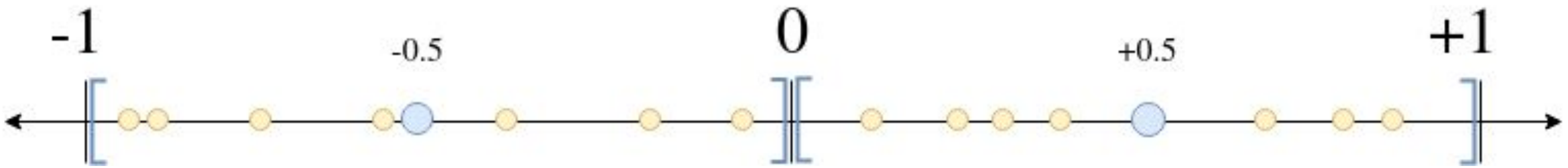


Binarized Neural Network



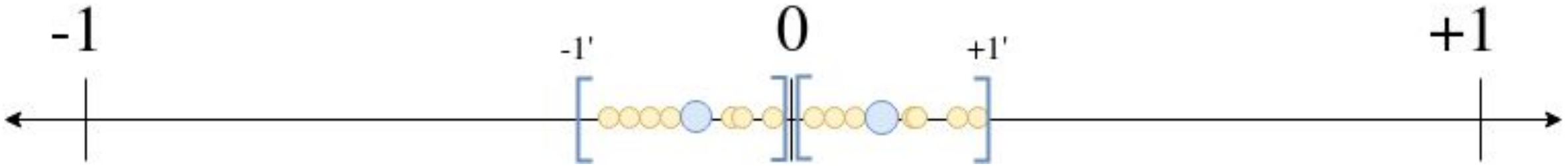
Proposed Methodology

Minimizing Variance



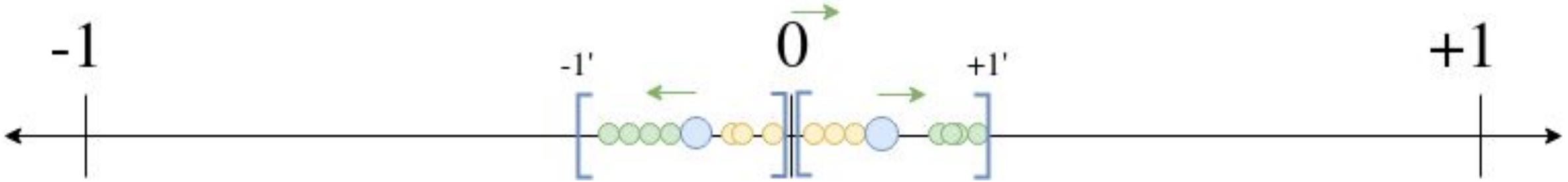
Proposed Methodology

Xavier Initialization



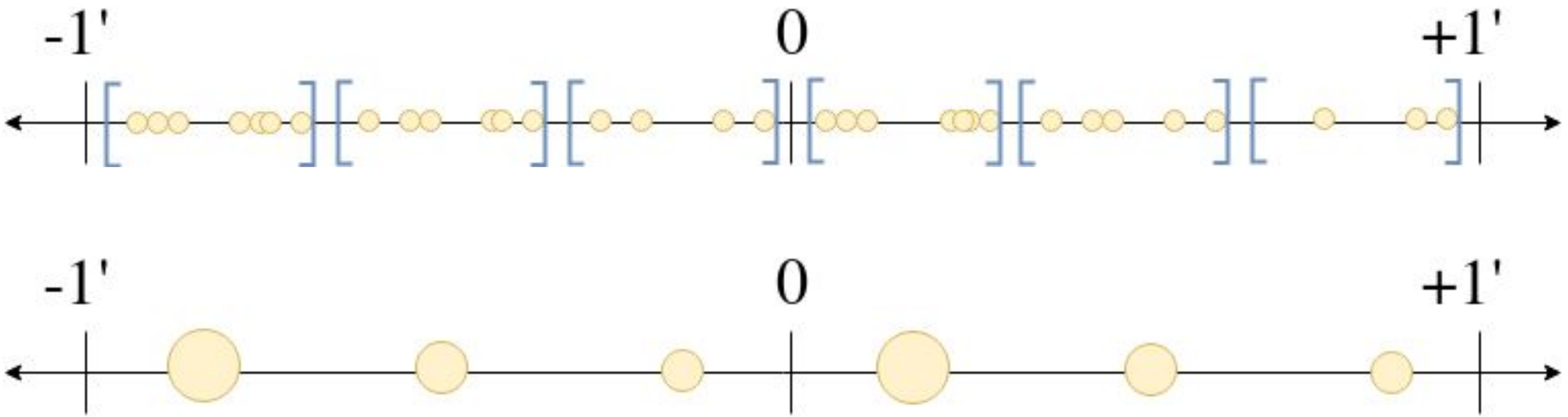
Proposed Methodology

Thresholding



Proposed Methodology

Bucketing



Implementation Details

- Dataset: AG_NEWS [8]
- Programming Language: Python
- Framework: PyTorch
- Library: Hugging face - BERT Tokenizer [9]
- GPU Language: Cuda
- GPU: RTX 3060
- RAM: 16G , VRAM: 6G

We constructed a transformer model with two encoder layers for text classification task as our baseline.

For all models on all experiments, we maintained the same training setting and trained for 10 epochs for comparability.

Evaluation Metrics

1. **Accuracy:** Measures how often a model predicts a correct outcome. Higher accuracy indicates to better performance.

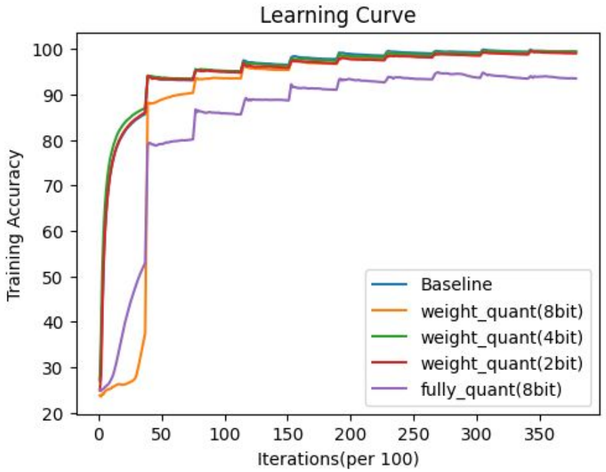
2. **Model Size:** Storage which we get by multiplying the number of parameters with the precision bytes. Lower model size means reduced memory footprint and potentially faster inference times.

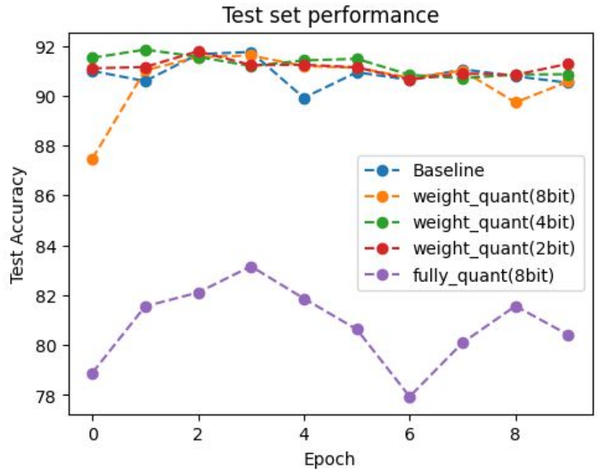
3. **Reduction Ratio:** The base model size divided by the compressed model size. The higher the ratio, the better the compression.

$$Reduction\ Ratio = \frac{Original\ size\ - Compressed\ size}{Original\ size} \times 100\%$$

Performance under several cases

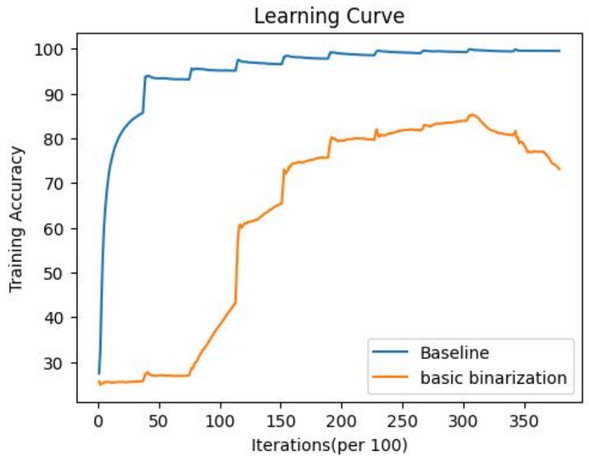
Experiment 1: Effectiveness of different quantization methods

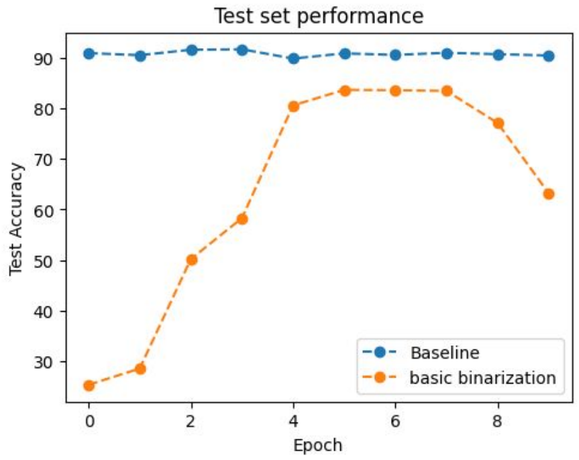




Model	Accuracy	Model Size	Reduction Ratio
baseline	91.8	78.7M	-
8-bit quant	91.6	38.6M	50.95 %
4-bit quant	91.8	19.7M	74.96 %
2-bit quant	91.8	10.2M	87.03 %
fully quant	83.2	38.6M	50.95 %

Table 1: Quantization Method



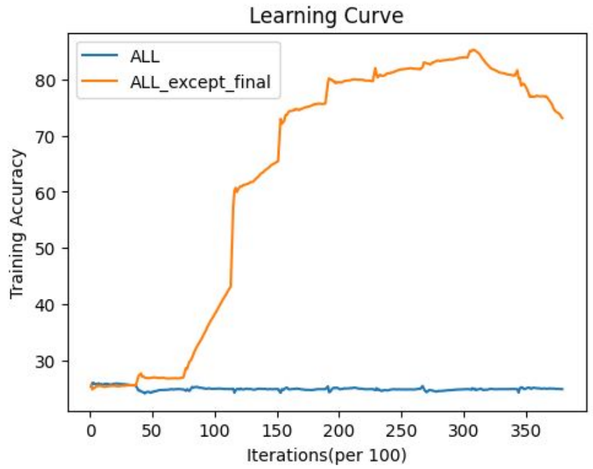


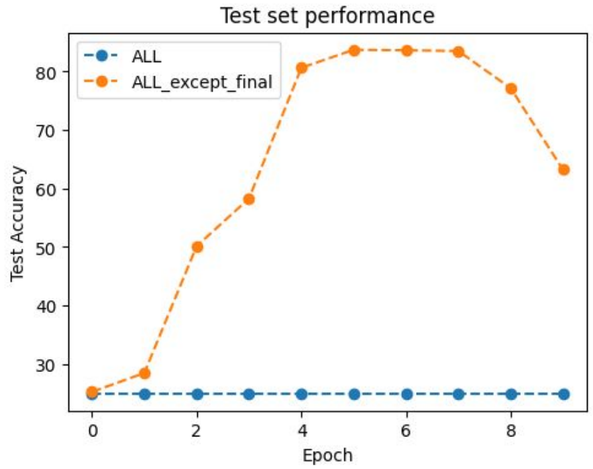
Model	Accuracy	Model Size	Reduction Ratio
baseline	**91.8**	78.7M	-
basic binarization	83.7	4.7M	94.03 %

Table 2: Binarization Method

Performance under several cases

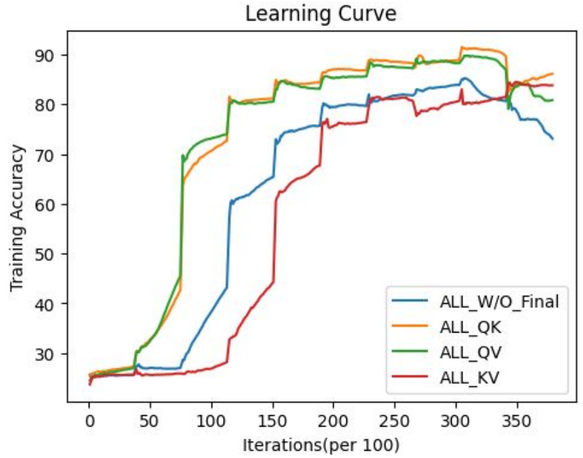
Experiment 2: Sensitivity of different parts of transformer

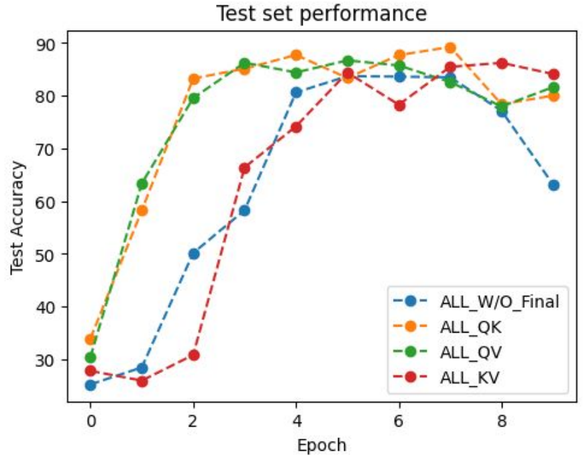




Model	Accuracy	Model Size	Reduction Ratio
all	25.0	4.7M	94.03 %
all except final	**83.7**	4.7M	94.03 %

Table 3: Effect of Final Layer

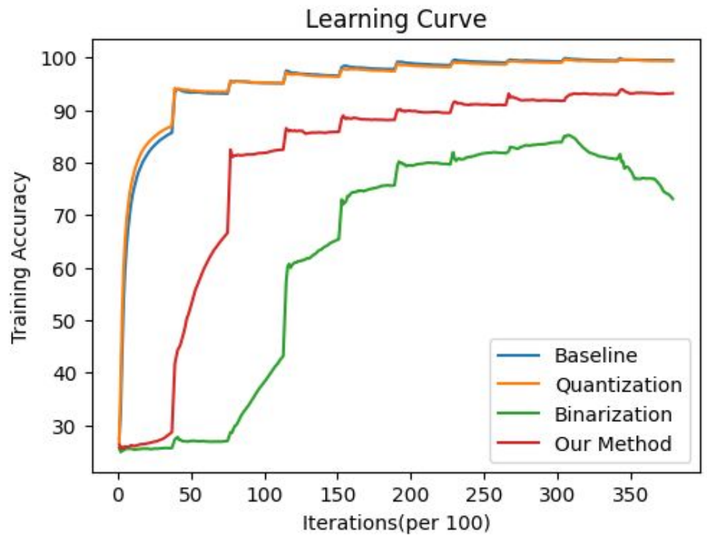


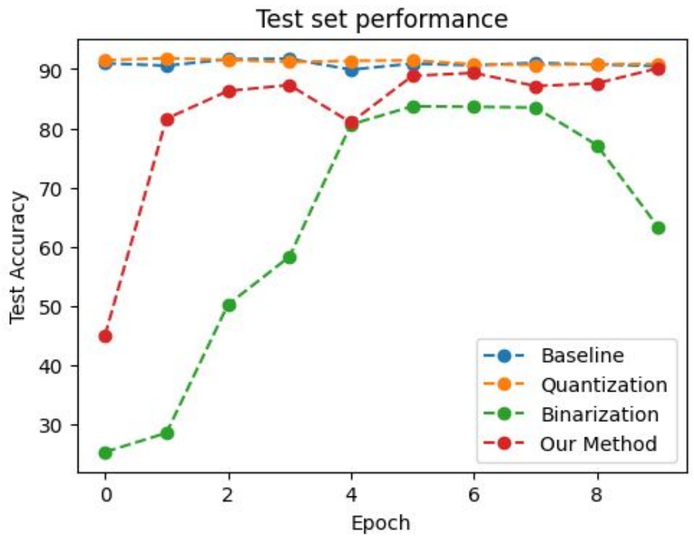


Model	Accuracy	Model Size	Reduction Ratio
all except final	83.7	4.7M	94.03 %
all (QK) except final	**89.3**	22.8M	71.03 %
all (QV) except final	86.7	22.8M	71.03 %
all (KV) except final	86.2	22.8M	71.03 %

Table 4: Effect of Binarized QK vs QKV

Comparison with the existing solutions





We can see that our method -

- ➢ Having an accuracy of 90.1 which is close to baseline.
- ➢ Having a lower model size than baseline.
- ➢ Having a higher reduction ratio.

Model	Accuracy	Model Size	Reduction Ratio
baseline	**91.8**	78.7M	-
4 bit quant	**91.8**	19.7M	74.96 %
binarization	83.7	4.7M	94.03 %
Our method	90.1	**4.7M**	**94.03 %**

Table 5: Our Method vs Existing Optimization Methods

- ➢ Having better compression than Quantization and better accuracy than Binarization

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[2] Partha Pratim Ray. Chatgpt: A comprehensive review on background, applica- tions, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems*, 2023.

[3] Zhuohan Li, Eric Wallace, Sheng Shen, Kevin Lin, Kurt Keutzer, Dan Klein, and Joey Gonzalez. Train big, then compress: Rethinking model size for efficient training and inference of transformers. In *International Conference on machine learning*, pages 5958–5968. PMLR, 2020.

[4] Brown, Nathan, Ashton Williamson, Tahj Anderson, and Logan Lawrence. "Efficient Transformer Knowledge Distillation: A Performance Review." *arXiv preprint arXiv:2311.13657* (2023).

[5]] Xiaoxia Wu, Cheng Li, Reza Yazdani Aminabadi, Zhewei Yao, and Yuxiong He. Understanding int4 quantization for language models: latency speedup, compos- ability, and failure cases. In *International Conference on Machine Learning*, pages 37524–37539. PMLR, 2023.

[6] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. *Advances in neural information processing systems*, 29, 2016.

[7] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[8] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *NIPS*, 2015.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre- training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.