NYU

From BERT to Mamba: Evaluating Deep Learning for Efficient QA Systems

Group SQuAD Squad

12.05.24

INTRODUCTION

Objective: Evaluate the performance of diverse deep learning models (**BERT**, **T5**, **LSTM**, **Mamba**) for a QA-based NLP task, aiming to balance **accuracy** and **computational efficiency**.

Focus:

- Fine-tune models on the **SQuAD 2.0 dataset** to extract meaningful QA insights.
- Analyze **Exact Match scores** and **resource utilization** for each architecture.
- Investigate trade-offs in accuracy and efficiency for **practical QA system deployment**.

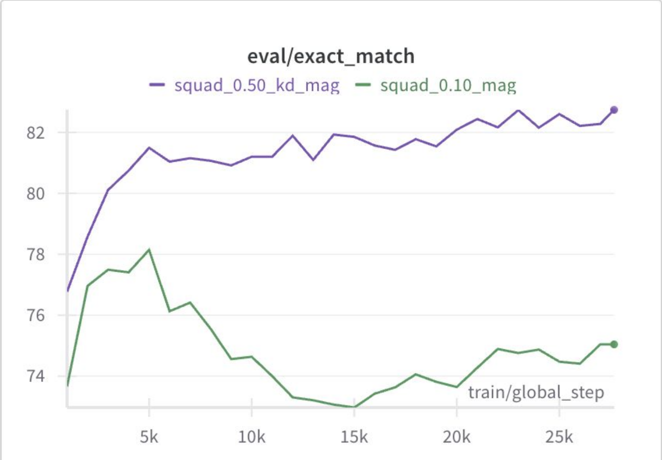
Architecture Overview:

- **BERT**: Encoder-only Transformer, excels at contextual embeddings.
- **T5**: Encoder-decoder architecture, versatile for text-to-text tasks.
- **LSTM**: Sequential model, effective for capturing dependencies.
- **Mamba**: State-space model optimized for **efficient long-sequence processing**.

BERT
(Bidirectional Encoder Representations from Transformers)

A transformer-based model utilizing an encoder-only architecture, designed to capture deep contextual embeddings for effective natural language understanding.

	Bert-Base-Uncased + Fine Tuning	Static Model Pruning	Static Model Pruning + KD
Remaining Weights	100%	10%	50%
Model Size	~110 million	~85 million	~85 million
Accuracy (EM)	0.65	0.75	0.82



- **Frozen Model Weights**: The model weights were kept frozen to avoid fine-tuning, preserving the original parameters.
- **Efficient Pruning**: A binary mask was learned to enable efficient pruning, eliminating the need to update the original weights.

NYU

T5
(Text-to-Text Transfer Transformer)

T5 is a generative LLM model with encoder-decoder architecture to process and generate text, where every NLP task is framed as a text-to-text problem.

	T5	T5 + LoRA	T5 + Quantization	T5 + QLoRA
Parameters	220M	220M + 800K	220M	220M + 800K
Trainable Parameters	220M (100%)	800K (0.3%)	220M (100%)	800K (0.3%)
Model Size	~850MB	~850MB	~212MB	~212MB
Accuracy (EM)	0.74	0.71	0.72	0.69

- **Quantization** lowers the precision of weights and activations (e.g., FP32 to INT8)
- **LoRA** reduces the number of trainable parameters by injecting lightweight, low-rank adapters into the model.

LSTM
(Long Short Term Memory)

A type of Recurrent Neural Network (RNN) designed to capture sequential patterns effectively but struggled with long-range dependencies

Feature	Model_1	Model_2	Model_3	Model_4
Embeddings	Random	GloVe (pre-trained)	FastText (pre-trained)	FastText (300 dim)(pre-trained)
Attention Mechanism	Attention	Attention	Multi-Head Attention	Self- attention
LSTM Configuration	Bidirectional	Bidirectional + Regularized	Bidirectional + Residual	Bidirectional + Residual
Pooling	Flatten	Flatten	GlobalMaxPooling1D	GlobalMaxPooling1D
Regularization	Dropout	Dropout, L2	Dropout, L2	Dropout, L2
Accuracy (EM)	0.093	0.029	0.116	0.22

NYU

MAMBA

Mamba is a new hardware aware LLM architecture that builds upon the Structured State Space sequence (S4) model to manage lengthy data sequences.

Pros: Selectively remembers information, higher throughput, simpler architecture.
Cons: Selectively remembers information, suffers from tail saturation, sequential inputs only.

Model architecture / Environment	Prompting
• Base Model: Mamba-130M (24 layers, 768 dim) • We use mamba-ssm with convd for optimized SRAM storage on CUDA V100 GPUs • Configuring SFT Trainer (transformers library) for fine-tuning along • GPT-NeoX tokenizer with special handling for EOS/PAD tokens	• Balancing the dataset with positive/negative samples generated. • Synthetic negative samples are generated by pairing random questions with “I don’t know” as the answer. • Using N-shot prompting to establish a consistent format. Using 3-5 samples provides a considerable improvement on factual questions.

With N-shot prompting, we achieved an EM accuracy of 0.12 using the Mamba-130M model.

Summary of Insights

- BERT : Static Model Pruning effectively reduces computational overhead by learning only a binary mask without fine-tuning the pre-trained weights. This approach prunes a significant portion of the model (e.g., 50%) while maintaining high accuracy, demonstrating its ability to save time and resources while achieving strong performance with fewer parameters.
- T5 : Combining LoRA and quantization significantly reduces the T5-base model's size (by 75%) and trainable parameters (by 99.97%) while achieving an efficient fine-tuning process with insignificant drop in the exact match score.
- LSTM : Using Self attention and residual connections, the LSTM model achieves the best performance when trained with FastText embeddings compared to Random and GloVe embeddings.
- Mamba : Despite the seemingly revolutionary new architecture, we see Mamba performing quite poorly. Here’s why we think this happened:
    - It’s pretraining step isn’t specifically designed for QA tasks like that of T5 or BERT
    - Mamba suffers from being on the efficiency side of the *efficiency vs. effectiveness* tradeoff faced by various LLM architectures. The lack of self attention means that Mamba can completely forget important bits of information from the context and perform poorly.

THANK YOU