Will The Real SQL Translator Please Stand UP?

Tyler Cranmer, Jayant Duneja

University of Colorado Boulder

Introduction

- The proposal aims to develop an interface that simplifies accessing Electronic Health Records (EHR) data by converting natural language questions into SQL queries using a text-to-SQL model.
- The approach includes leveraging Small Language Models (SLMs) with fewer parameters and advanced fine-tuning techniques to potentially enhance performance and cost efficiency.
- The dataset we used comes from the NAACL - Clinical NLP 2024 shared task, which comprises 6,291 natural language instruction queries and their expected SQL queries for both training and testing.

Motivation

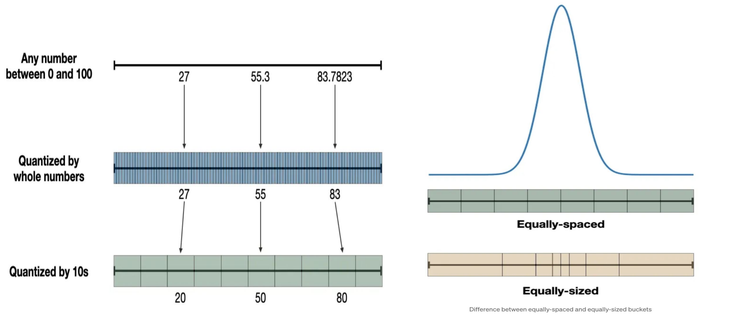
- As Generative LLMs are becoming more and more popular and they are being used by a larger share of the population, privacy concerns around these models have also increased.
- Fine-tuning smaller language models (SLMs) for smaller, more specific tasks can help replicated the performance we get from LLMs.
- Having in-house fine-tuned models help getting past the privacy concerns around these models.

Quantization

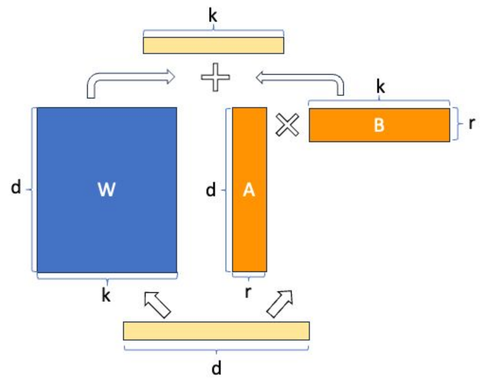
- The quantization technique in QLoRA refers to quantizing the precision of the weight parameters in the pre-trained LLM to 4 bit precision, which are typically stored in a 32 bit format.
- This reduces the memory footprint of the model, making it possible to train it on a single GPU.

Model	Fine-tuning Memory	Quantized Memory
Llama 2 70B	~1100GB	~140GB
Llama 3 70B	~1100GB	~140GB
Llama 2 7B	~110GB	~28GB
Llama 3 8B	~125GB	~37GB

* Memory may vary based on training batch size and quantization parameters.



LoRA (Low Rank Adaptation)



- LoRA is a technique that accelerates the fine-tuning of LLMs, by decomposing their weight matrices into two, smaller low rank matrices.
- These new matrices can be trained to adapt the new data while the original model weights remain frozen.
- For inference, both the model and the adapter weights are combined.
- LoRA makes fine-tuning more efficient by drastically reducing the number of trainable parameters.

Training Data

Training data set consisted of 5,124 original examples and 20,539 augmented examples.

Original natural language instruction:

- *Tell me the minimum respiratory rate in patient 10021118 in the first ICU visit.*

Augmented natural language instruction:

- *Provide the minimum respiratory rate recorded for patient 10021118 on their first ICU admission*
- *Can you report the lowest respiratory rate observed for patient 10021118 during their first visit to the ICU?*
- *Please inform me of the minimum respiratory rate recorded for patient 10021118 during their first stay in the ICU.*

Input Prompts

The below section illustrates how the training and the inference data was formatted for LLama-2 and Llama-3.

Training

Llama-2:

```
<s>[INST] <<SYS>>
{{ system_prompt }}
<</SYS>>
{{ user_message }} [/INST]
{{ model_response}}</s>
```

Llama-3:

```
<|begin_of_text|>
<|start_header_id|>system<|end_header_id|>
{{ system_prompt }}<|eot_id|>
<|start_header_id|>user<|end_header_id|>
{{ user_ message }}<|eot_id|>
<|start_header_id|>assistant<|end_header_id|>
{{ model_response }}<|eot_id|>
<|end_of_text|>
```

Inference

Llama-2:

```
<s>[INST] <<SYS>>
{{ system_prompt }}
<</SYS>>
{{ user_message }} [/INST]
```

Llama-3:

```
<|begin_of_text|>
<|start_header_id|>system<|end_header_id|>
{{ system_prompt }}<|eot_id|>
<|start_header_id|>user<|end_header_id|>
{{ user_ message }}<|eot_id|>
<|start_header_id|>assistant<|end_header_id|>
```

Loss and Attention Masks

- Attention Mask: Indicates the tokens which the model should pay attention to.
- Loss Mask: Indicates the tokens the model should learn from, i.e be punished for generating incorrect results.

Input Structure:
[<s>, [INST], <<SYS>>, System Prompt, <</SYS>>, Health, Professional, Query, [/INST], Generated, SQL, Query, </s>, |<pad>|, |<pad>|, |<pad>|]

Attention Mask:
[1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0]
The attention mask uses 1 for actual tokens and 0 for padding, guiding the model's focus during processing.

Loss Mask:
[0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0]
The loss mask targets the "Generated SQL Query" tokens with 1 for learning emphasis, ignoring other tokens and padding with 0.

Example displaying the Loss and Attention masks for a particular model input. (Actual tokens may differ from this example, have simplified it for user understanding)

Results

We are using the metrics which have been specified by the shared task organizers.

We have also implemented a small post-processing script on top of our original results when the generated SQL query was leading to an error on running against the database

Model	Accuracy %	Accuracy % after Post-Processing
GPT-3.5*	27.0	28.3
GPT-4*	30.7	31.6
Llama 2 7B (Fine Tuned)	31.8	51.5
Llama 3 8B (Fine Tuned)	59.6	60.6

* The GPT-4 and GPT-3.5 results were generated using the prompt format specified as the baseline by the organizers. We have not implemented any prompt formatting or RAG techniques with the GPT models.

References

1. Llama Documentation : https://llama.meta.com/docs/get-started
2. Edward Hu et al. (2021). “Low Rank Adaptation of Large Language Models”. https://arxiv.org/pdf/2106.09685.
3. Hugging Face. “Anatomy of Models Memory.” Accessed April 29, 2024. https://huggingface.co/docs/transformers/perf_train_gpu_one#anatomy-of-models-memory.