Análise Quantitativa do Trade-off entre Especialização e Generalização em LLMs via Fine-Tuning

Manoel Victor Florencio de Souza
ICOMP - UFAM
Manaus, Brasil
manoel.souza@icomp.ufam.edu.br

I. Metodologia

Esta seção irá apresentar uma descrição do *pipeline* de processamento de dados, configuração do modelo utilizado, do LoRA utilizado para treinamento do modelo e da arquitetura da métrica para avaliação dos modelos na tarefa de *Text-to-SQL*.

A. Pipeline de processamento de dados

Para este trabalho foram utilizados 2 bases de dados, sendo elas a *MMLU* que possui questões de diversos campos do conhecimento como Astronomia, Econometria e Religiões do Mundo entre outros e o *Spider* que trata de questões em texto livre a serem respondidas com consultas SQL dado um banco de dados.

1) MMLU: Dados 3 categorias gerais STEM, Humanidades e Ciências Sociais foram escolhidos as sub-categorias Astronomia, Econometria e Religiões do Mundo, respectivamente e selecionadas 50 questões do sub-conjunto de teste de cada sub-categoria a fim de construir um questionário final para avaliação de variação de desempenho dos modelos após ajuste fino para uma tarefa específica. A seleção das questões foi feita a partir do conjunto de teste de cada sub-categorias embaralhadas com semente fixa de 42 e então extraídos os 50 primeiros exemplos.

Para construção do prompt de resposta foram selecionados 4 casos dos sub-conjuntos de treinamento do MMLU das 3 sub-categorias especificadas anteriormente e mais uma sub-categoria não relacionada (philosophy) a fim de construir um prompt 4-shot com exemplos das 4 alternativas de resposta.

2) Spider: Da base Spider foi utilizada a base de dados de treinamento final oficial (train_spider.json e train_others.json) que consiste de 8.659 triplas (esquema do banco de dados, pergunta, resposta em SQL que responde a pergunta) a fim de treinamento dos modelos de linguagem escolhidos e a base de dados de desenvolvimento (dev.json) para teste dos modelos.

Para construção do prompt de resposta foram selecionados aleatoriamente 3 casos da base de dados de treinamento para se ter um prompt 3-shot, a seleção foi manual considerando 3 complexidades de consulta, as consultas usadas foram excluídas da base de treinamento final. O prompt consiste numa breve explicação da tarefa alvo bem como da entrada esperada, um detalhamento do esquema do banco de dados, a pergunta em texto natural e a consulta SQL esperada.

B. Configuração dos modelos de linguagem utilizados

Para os experimentos foram utilizadas 3 configurações do modelo *LLaMa 3 8b Instruct*, a primeira sendo o modelo base sem ajuste fino, a segunda o modelo com ajuste fino e taxa de aprendizado em $1e^{-4}$ e a terceira o modelo com ajuste fino e taxa de aprendizado em $5e^{-5}$.

Como configurações de quantização foi utilizada uma quantização em 4 bits em *bfloat16*, quantização dupla e tipo de quantização em *nf4*.

As configurações baseadas em ajuste fino tiveram seus hiper-parâmetros LoRA estáticos, sendo eles $r = 8$, $\alpha = 16$, $dropout = 0.05$, sem viés, e módulos alvo q_proj, o_proj, k_proj, v_proj, gate_proj, up_proj e down_proj.

Como argumentos de treinamento foram utilizados 1 época de treinamento, tamanho de batch por dispositivo igual a 1, passos de acumulação de gradiente igual a 4, Adam Ponderado como otimizador, decaimento de pesos igual a 0.001, normalizaçaõ máxima de gradiente igual a 0.3, proporção de passos para aquecimento da taxa de aprendizado igual a 0.03, cosseno como tipo de *scheduler* da taxa de aprendizado, fora as taxas de aprendizado $1e^{-4}$ e $5e^{-5}$ expostos anteriormente como variações dos modelos.

Os dados de treinamento foram divididos com 80% para treinamento real e 20% para validação, os dados foram embaralhados com semente fixa de 42 e então separados 80% exemplos iniciais para treino e 20% finais para validação.

C. Arquitetura da métrica ExecutionAccuracy

Para cálculo da métrica foram verificados os resultados das consultados reais e das consultas construídas pelos modelos, se houvesse igualdade entre respostas o resultado era ”1.0” e ”0.0” em caso negativo.

II. Resultados

Esta seção será dividida em 3 partes, a primeira irá apresentar o desempenho do modelo base na tarefa de Text-to-SQL na base de dados de desenvolvimento do Spider e os ganhos nesta tarefa após ajuste fino do modelo nas duas configurações descritas anteriormente, na segunda irei apresentar 2 exemplos

de cada configuração onde os modelos ajustados falharam na tarefa de Text-to-SQL e por fim uma comparação do desempenho do modelo base e ajustados na tarefa de resposta a perguntas do MMLU nas 3 sub-categorias descritas na sub-seção I-A1.

A. Desempenho dos modelos na tarefa de Text-to-SQL

O modelo base foi submetido aos 899 prompts da base de dados de desenvolvimento do Spider, em seguida foram realizados os ajustes finos, com adaptadores ajustados salvos os modelos foram submetidos aos mesmo 899 prompts e seus resultados foram comparados baseado na métrica *Execution-Accuracy* explicitado na sub-seção I-C, os resultados podem ser vistos na Tabela I.

TABLE I
MÉTRICAS ALCANÇADAS

Modelo	ExecutionAccuracy (%)
LLaMa 3 8b Instruct base	41.27
LLaMa 3 8b Instruct LR$1e^{-4}$	46.16
LLaMa 3 8b Instruct LR$5e^{-5}$	**55.17**

Em ambos os casos de ajuste fino para a tarefa alvo houve ganhos no desempenho, sendo de 4.89% da configuração com taxa de aprendizado em $1e^{-4}$ como de 13.90% para a configuração com taxa de aprendizado em $5e^{-5}$, mostrando que a construção dos prompts e treinamento do modelo foi efetiva e deixa espaço para melhorias, sendo por mais épocas de treinamento, melhor processamento da base de dados ou mais testes de hiperparâmetros.

B. Análise de erros dos modelos ajustados na tarefa de Text-to-SQL

Entre as perguntas usadas para testes dos modelos contém a pergunta "*List the maximum weight and type for each type of pet.*", neste caso os 3 modelos erraram quando vendo a resposta original, sendo:

- Consulta real: 'SELECT max(weight) , petType FROM pets GROUP BY petType'
- LLaMa base: 'Here is the SQL answer:SELECT PetType, MAX(weight) FROM Pets GROUP BY PetType;'
- LLaMa LR $1e^{-4}$: 'SELECT PetType, MAX(weight) AS MaxWeight, PetID FROM Pets GROUP BY PetType To answer the question, we use the GROUP BY clause to group the pets by their type, and the MAX function to find the highest weight for each group. We also include the PetID column in the SELECT clause to preserve the original table structure.'
- LLaMa LR $5e^{-5}$: 'SELECT PetType, MAX(weight) FROM Pets GROUP BY PetType'

Nos casos do LLaMa base e LLaMa LR $5e^{-5}$ é possível ver que a consulta tem sentido e utiliza das colunas e cláusulas corretamente, porém erram devido a sensibilidade de caixa alta do SQL, mesmo limpando a saída para remover o *reasoning* ainda haveria erro sem adaptação desses pontos. No caso do LLaMa LR $1e^{-4}$ acontece de errar mesmo desconsiderando sensibilidade de caixa alta e *reasoning* pois inclui a coluna PetID que altera o sentido da busca.

Em outro caso a pergunta "*Find the id of the pet owned by student whose last name is 'Smith'.*", o modelo base e o modelo LR $5e^{-5}$ acertaram porém o modelo LR $1e^{-4}$ errou, sendo:

- Consulta real: "SELECT T2.petid FROM student AS T1 JOIN has_pet AS T2 ON T1.stuid = T2.stuid WHERE T1.Lname = 'Smith'"
- LLaMa base: "SELECT PetID FROM Has_Pet JOIN Student ON Has_Pet.StuID = Student.StuID WHERE Student.LName = 'Smith';"
- LLaMa LR $1e^{-4}$: "SELECT T1.StuID FROM Student T1 WHERE T1.LName = 'Smith'This SQL answer selects the StuID from the Student table where the LName is 'Smith', answering the question by finding the ID of the pet owned by the student with the last name 'Smith'."
- LLaMa LR $5e^{-5}$: "SELECT PetID FROM Has_Pet WHERE StuID IN (SELECT StuID FROM Student WHERE LName = 'Smith')"

Mesmo desconsiderando o *reasoning* do modelo LR $1e^{-4}$ a consulta ainda não entrega o que foi solicitado na pergunta.

Em um terceiro caso, foi feita a pergunta '*What are the different countries with singers above age 20?*' e apenas o modelo base acertou, com os modelos ajustados errando a semântica da pergunta mesmo desconsiderando *reasoning*.

- Consulta real: 'SELECT DISTINCT country FROM singer WHERE age $>$ 20'
- LLaMa base: 'SELECT DISTINCT s.Country FROM singer s WHERE s.Age $>$ 20;'
- LLaMa LR $1e^{-4}$: 'SELECT T1.Country FROM Singer AS T1 WHERE T1.Age $>$ 20 GROUP BY T1.Country This SQL answer will provide the different countries with singers above age 20 by grouping the countries from the Singer table where the age is greater than 20.'
- LLaMa LR $5e^{-5}$: 'SELECT T1.Country FROM Singer AS T1 WHERE T1.Age $>$ 20 GROUP BY T1.Country (Note: The answer does not require a join because the question is asking for different countries with singers above age 20, which can be answered by looking at the singer table directly.)'

C. Desempenho dos modelos na tarefa de resposta a perguntas

Nesta sub-seção será discutido os resultados dos 3 modelos na base de teste de 150 resposta a perguntas do MMLU, a fim de verificar possíveis impactos que o ajuste fino pode ter causado ao modelo em outra tarefa.

TABLE II
PERFORMANCE DOS MODELOS NA TAREFA RESPOSTA A PERGUNTAS

Modelo	Acurácia (%)			
	STEM	Humanidades	Ciências Sociais	Agregado
Base	**64.0**	34.0	**90.0**	**62.67**
LR $1e^{-4}$	54.0	**48.0**	68.0	56.67
LR $5e^{-5}$	42.0	40.0	78.0	53.33

A partir da Tabela II é possível verificar que o impacto do ajuste fino na tarefa de Text-to-SQL não necessariamente degradou o modelo de forma homogênea para outra tarefa visto que o modelo Base obteve uma performance pior na categoria Humanidades do que os modelos ajustados, mesmo que na avaliação agregada o modelo Base ainda tenho se saído melhor.

Ainda é possível notar que que entre os modelos ajustados ainda não houve um que fosse melhor nas 3 categorias já que o modelo LR $1e^{-4}$ teve uma performance pior em na categoria Ciências Sociais do que o modelo LR $5e^{-5}$, ou seja, o impacto do ajuste fino foi heterogêneo entre domínios para os casos analisados.

III. Discussão

Nesta seção serão considerados alguns pontos de atenção vistos nos resultados obtidos durante o desenvolvimento deste trabalho, como o impacto do ajuste fino na tarefa alvo e a degradação do modelo após este ajuste fino para outras tarefas, fatores impactantes no ganho performance e implicações práticas destes achados.

Primeiramente, analisando os resultados na tarefa alvo pode-se dizer que o impacto do ajuste fino foi positivo visto que ambos os modelos ajustados tiveram performance superior ao modelo base, a perda de performance em uma tarefa adjacente pode ser vista como aceitável se for especializar o modelo de linguagem a depender do grau da perda, levando em consideração tarefas muitos específicos e a manutenção da habilidade de conversação do modelo pode-se criar modelos de linguagens para fins específicos se estes forem aplicados a apenas estas tarefas, ainda é possível especializar modelos de linguagem para fins específicos e manter um que seja mais geral para orquestra-los no caso da necessidade de manter o aspecto geral do modelo.

Em relação ao desenvolvimento de LLMs comerciais especializados, é necessário verificar as necessidades gerais e específicas para as atividades que essas LLMs irão auxiliar, garantindo veracidade nas informações, assertividade de resposta e tratamento de erros quando ocorram.

Nenhum treinamento ou especialização de modelos irá garantir um funcionamento pleno e sem necessidade de validação humana sobre as tarefas em que essa LLM será aplicada, portanto mesmo que o treinamento faça a LLM atingir um nível aceitável de performance para uma tarefa ainda será necessário a supervisão humana ao longo do tempo para aferir degradação do modelo.