On Finetuning Large Language Models

Resumo:

O artigo analisa a alegação feita de que uma abordagem baseada em dicionários interpretáveis (OCoDi), combinada com modelos como Random Forest e XGBoost, supera modelos de linguagem como o BERT ajustado (finetuned).

Mostra que essa conclusão é enganosa porque os autores ajustaram apenas a camada de classificação do BERT, congelando todos os outros milhões de parâmetros — o que limita muito a capacidade do modelo.

Ao permitir que todos os parâmetros do BERT sejam ajustados, demonstra que:

- O modelo aprende mais rapidamente,
- Tem erro quadrático médio 46% menor,
- E um R^2 27% maior do que os modelos baseados em dicionário.

Além disso, ao aumentar o tamanho máximo da sequência de entrada de 256 para 512 tokens, o desempenho melhora ainda mais.

Embora esse ajuste completo leve mais tempo para treinar, o custo computacional continua viável. Assim, o artigo conclui que, quando bem ajustado, o BERT supera abordagens interpretáveis, mesmo que estas últimas sejam mais fáceis de entender.

Understanding parameter-efficient Finetuning of Large Language Models: From Prefix Tuning to LLaMa-Adapters

Resumo:

Com o crescimento dos grandes modelos de linguagem (LLMs) como GPT e BERT, adaptar esses modelos para tarefas específicas tornou-se essencial. Porém, o ajuste fino completo (finetuning) pode ser extremamente caro e inviável em termos de tempo, memória e energia. Para resolver esse problema, surgiram técnicas de ajuste fino eficiente em parâmetros (PEFT – Parameter-Efficient Fine-Tuning).

Conceitos Centrais

1. Ajuste fino tradicional:
    - Feature-based: extrai embeddings com um LLM congelado e treina um classificador externo (ex: regressão logística).
    - Finetuning I: congela o LLM e ajusta apenas as camadas de saída.
    - Finetuning II: ajusta todas as camadas do modelo, o que normalmente oferece melhor desempenho, mas é muito mais caro (ex: 110 milhões de parâmetros no BERT base vs. 1.500 na camada final).

Exemplo de acurácia com DistilBERT:

- Feature-based(Baseado em recursos): 83%
- Finetuning I (Finetuning nas camadas de saída, parâmetros do LLM pré-treinado congelados): 87%
- Finetuning II (Finetuning em todas as camadas, não congela os parâmetros do LLM pré-treinado, mas também os ajusta): 92%

Ajuste Fino Eficiente em Parâmetros (PEFT)

PEFT permite ajustar grandes modelos com apenas uma pequena fração dos parâmetros, reduzindo drasticamente os custos.

Técnicas populares:

1. Prompt Tuning e Prefix Tuning
   - Soft prompt tuning: adiciona vetores treináveis ao input.
   - Prefix tuning: adiciona tensores treináveis em cada bloco do transformer.
   - Usa apenas 0.1% dos parâmetros.
   - Pode superar o ajuste fino completo em tarefas pequenas por evitar overfitting.

2. Adapters
   - Camadas adicionais pequenas (tipo autoencoder) inseridas nos blocos transformer.
   - Treinam só ~3.6% dos parâmetros.
   - Levemente inferior ao prefix tuning em eficiência, mas também eficaz.

LLaMA-Adapter: Um Avanço PEFT

O LLaMA-Adapter introduz um mecanismo de atenção com inicialização zero acoplado a um sistema de gating para estabilizar o treinamento. Essa técnica visa evitar que prompts de prefixo ou camadas de adaptadores com tensores inicializados aleatoriamente prejudiquem o conhecimento linguístico já aprendido pelo LLM, o que poderia causar instabilidade e altas perdas no início do ajuste fino.

Diferente dos métodos tradicionais de prefix tuning e adapters, o LLaMA-Adapter aplica os prompts de adaptação apenas às camadas superiores do transformer, focando em representações semânticas de mais alto nível.

LLaMA-Adapter combina ideias de prefix tuning e adapters. Ele introduz:

- Prefixos aprendidos internamente via tabela de embeddings.
- Aplicação dos prefixos apenas nas últimas camadas do transformer.
- Atenção inicializada em zero + mecanismo de gating, para estabilizar o treinamento e evitar perturbar o conhecimento linguístico pré-treinado.

Resultados práticos:

- Ajustou um modelo LLaMA com 7 bilhões de parâmetros em apenas 1 hora (com 8 GPUs A100), usando só 1.2M parâmetros treináveis.
- Superou todos os outros modelos testados em tarefas de perguntas e respostas.

Conclusão

O ajuste fino completo traz alto desempenho, mas é custoso. Por isso, métodos como prefix tuning, adapters e LLaMA-Adapter surgiram como soluções que:

- Mantêm desempenho competitivo;
- Requerem apenas uma fração dos parâmetros;
- São viáveis em dispositivos com menos recursos;
- Reduzem custos energéticos e computacionais.

O LLaMA-Adapter é o exemplo mais recente e promissor de PEFT, com excelente desempenho e acessibilidade. O método Adapter proposto é um método geral que também pode ser aplicado a outros tipos de LLMs (como o GPT).