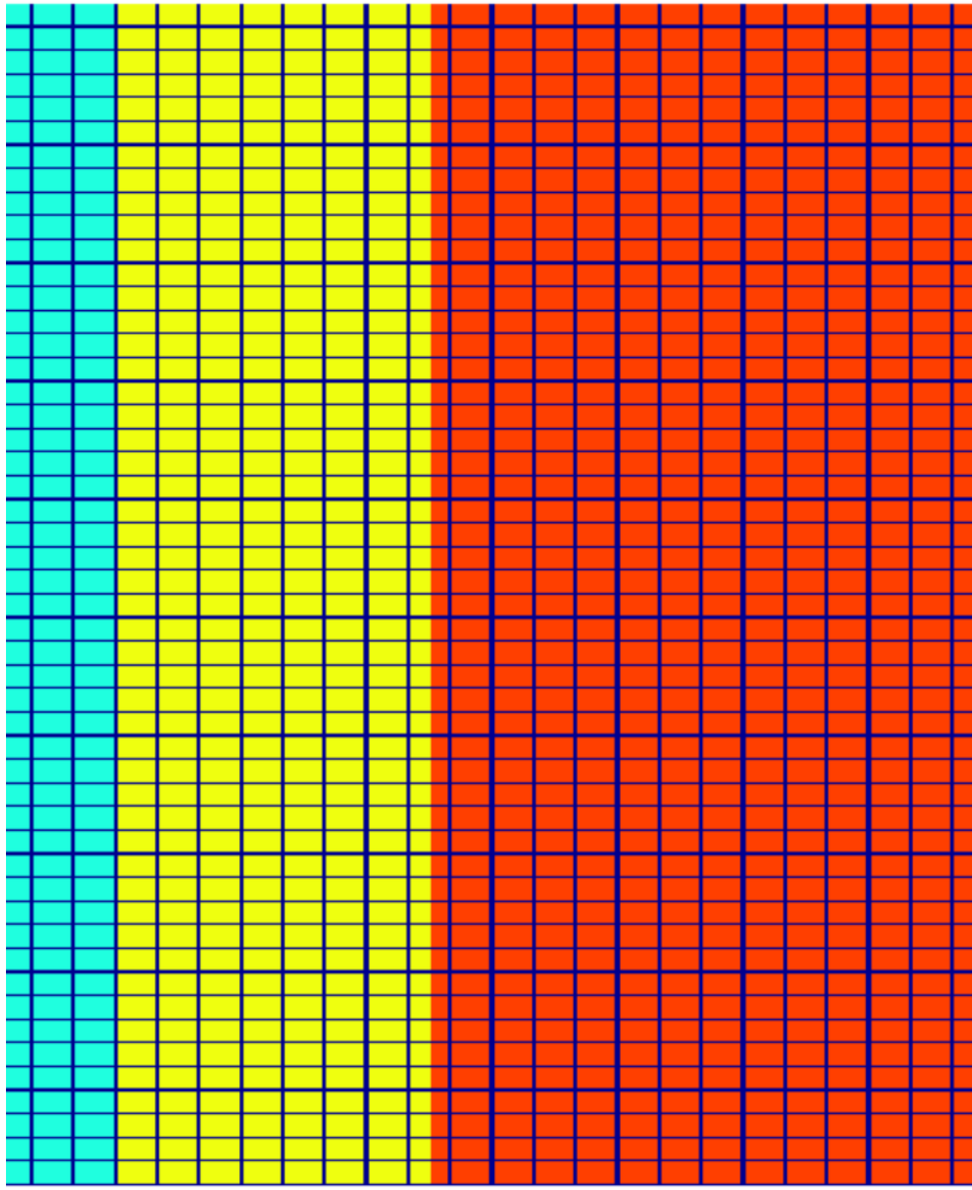
Ajuste fino com eficiência de parâmetros em modelos de linguagem: LoRA, QLoRA, RAG e PEFT

Nesta apresentação, conheceremos LoRA (Low-Rank Adaptation of Large Language Models), que é uma técnica revolucionária para adaptação de modelos de linguagem de alta capacidade. Entenderemos a ideia por trás do QloRA, RAG e PEFT. Veremos como o PEFT pode ser usado para ajustar o modelo para tarefas específicas de domínio, com o menor custo e infraestrutura mínima.



by Gerson Nascimento, Projeto5-LLM

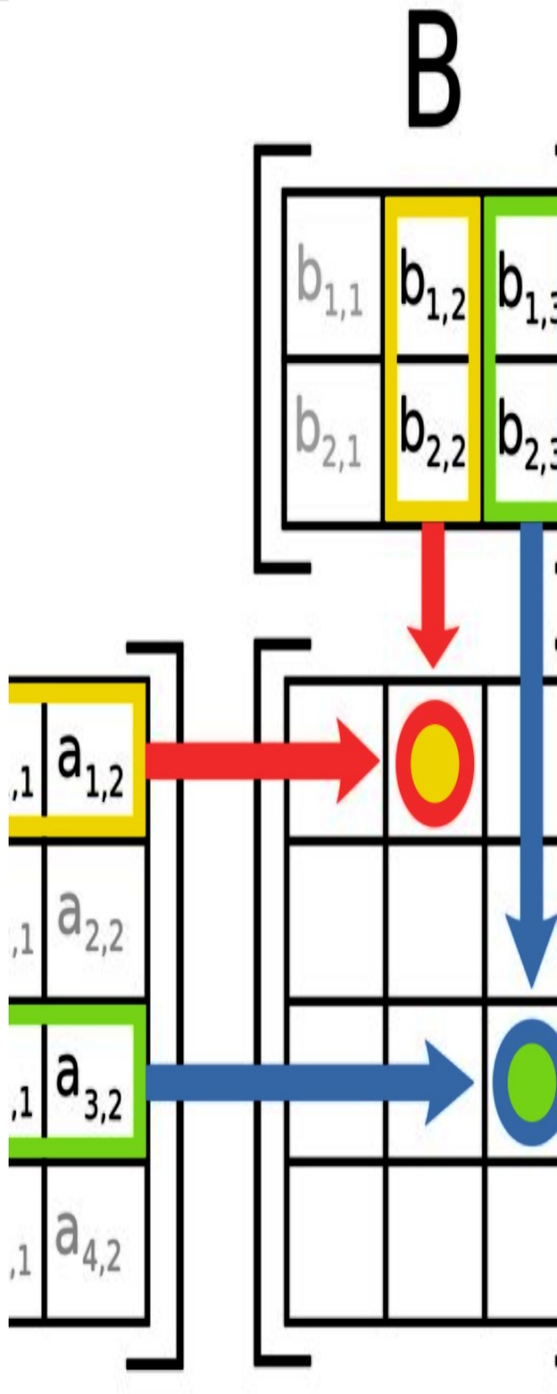
O que é LoRA?

LoRA é uma técnica para adaptar modelos de linguagem grandes a novos domínios, mantendo a capacidade preditiva e gerativa. LoRA usa uma fatorização de matrizes que permite um treinamento rápido e eficiente em dispositivos de baixa potência.

Decomposição SVD (Singular Value Decomposition)[redução de dimensionalidade]:
A decomposição SVD de uma matriz A é dada por: A = UΣ(V^T), onde U e V são matrizes ortogonais e Σ é uma matriz diagonal contendo os valores singulares de A.
U: Matriz de vetores singulares à esquerda.
Σ: Matriz de valores singulares.
V^T: Transposta da matriz de vetores singulares à direita.

Redução de rank [remover redundâncias]:
Suponha que a matriz A seja de dimensão (m x n), onde m representa o número de linhas e n o número de colunas. A matriz Σ é de dimensão (m x n), mas apenas os primeiros r valores singulares são significativos, onde r é o rank desejado. Podemos truncar a matriz Σ para ter dimensão (r x r), obtendo uma representação de menor dimensão.

LoRA:
Para aplicar o LoRA, consideramos a matriz de parâmetros do modelo de linguagem, que chamaremos de M. Aplicamos a decomposição SVD em M, obtendo: M = UΣ(V^T). Em seguida, truncamos Σ para ter dimensão (r x r), resultando em Σ_r. A matriz adaptada pelo LoRA é dada por: M' = UΣ_r(V^T).



Vantagens do uso de LoRA

1

Potencializa Modelos de Linguagem

LoRA permite que você aproveite a capacidade preditiva e gerativa de grandes modelos de linguagem para resolver tarefas específicas, eliminando a necessidade de treinar um novo modelo do zero.

2

Economiza Tempo e Recursos

Adaptar um modelo existente com LoRA é muito mais rápido e econômico do que treinar um modelo do zero. Isso permite que você desenvolva soluções mais rapidamente e com menor custo.

3

Melhora a Eficiência

Com LoRA, é possível reduzir a complexidade computacional dos modelos de linguagem, tornando-os mais eficientes em tarefas específicas, como tradução automática, resumo de textos e geração de texto.

Desvantagens do uso de LoRA

1

Dependência de Modelos Pré-treinados

LoRA requer um modelo de linguagem pré-treinado como ponto de partida, o que pode limitar as opções disponíveis e a flexibilidade nos resultados obtidos.

2

Complexidade de Implementação

A implementação de LoRA requer conhecimentos avançados em processamento de linguagem natural, álgebra linear e programação, o que pode ser um obstáculo para alguns desenvolvedores.

Passos para adaptação de LLMs com LoRA

1

Passo 1: Escolha do Modelo Base

Selecione um modelo de linguagem pré-treinado que seja adequado para a tarefa que você deseja resolver.

2

Passo 2: Coleta e Pré-processamento de Dados

Reúna dados relevantes e pré-processe-os para a tarefa específica.

3

Passo 3: Adaptação com LoRA

Utilize a técnica LoRA para ajustar os pesos do modelo, adaptando-o para a tarefa em questão.

4

Passo 4: Ajuste e Avaliação

Ajuste os hiperparâmetros do modelo adaptado e avalie seu desempenho. Faça ajustes adicionais, se necessário.

LoRA versus outras técnicas similares

Transfer Learning

O Transfer Learning permite aproveitar os conhecimentos prévios de um modelo para resolver tarefas diferentes. No entanto, LoRA é mais adequado quando se deseja adaptar um modelo para uma tarefa específica, mantendo sua capacidade preditiva e gerativa.

Fine-tuning

No Fine-tuning, todos os parâmetros do modelo são ajustados durante o treinamento. Em contraste, LoRA ajusta apenas uma parte reduzida, tornando-o mais eficiente e flexível em tarefas específicas.

Em resumo...

Decomposição SVD:
A decomposição SVD (Decomposição em Valores Singulares) é uma técnica que fatora uma matriz em três componentes: U, Σ e (V^T). Essa fatoração é usada para obter uma representação de menor dimensionalidade da matriz original.

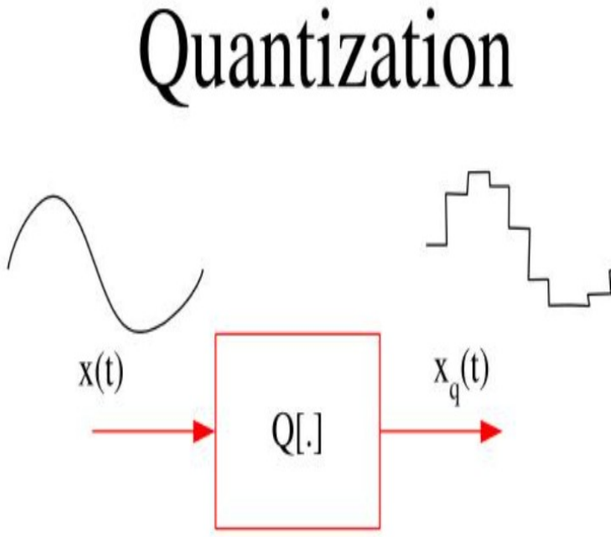
Redução de rank:
No LoRA, podemos reduzir o rank da matriz Σ para obter uma representação de menor dimensionalidade. Truncamos Σ para ter dimensão (r x r), onde r é o rank desejado.

Em resumo, o LoRA utiliza a decomposição SVD para reduzir a dimensionalidade do modelo de linguagem, preservando as informações relevantes e permitindo ajustar o modelo a novos contextos. Essa sustentação matemática baseada em decomposição de matrizes e teoria do baixo rank é o que torna o LoRA uma técnica eficaz para a adaptação de modelos de linguagem de grande escala.

QLoRA: Eficiente ajuste fino de LLMs quantizados

O QloRA (Efficient Finetuning of Quantized LLMs) é uma técnica avançada de ajuste fino para LLMs quantizados, que visa melhorar a performance e a eficiência dos modelos de linguagem. Ele utiliza métodos de otimização específicos para expandir a capacidade do algoritmo.

Entendendo os pilares...



Quantização:
A quantização é uma técnica que reduz a precisão numérica dos parâmetros de um modelo de linguagem, substituindo-os por valores discretos. Isso permite reduzir o tamanho do modelo e acelerar as operações.

Decomposição SVD:
A decomposição SVD (Decomposição em Valores Singulares) é uma técnica que fatora uma matriz em três componentes: U, Σ e (V^T). Essa fatoração é usada para obter uma representação de menor dimensionalidade da matriz original.

QLoRA:
No QLoRA, a matriz de parâmetros do modelo de linguagem quantizado é decomposta usando a SVD. Dessa forma, temos: M = UΣ(V^T), onde U e V são matrizes ortogonais e Σ é uma matriz diagonal contendo os valores singulares.

Redução de rank:
No QLoRA, assim como no LoRA, podemos reduzir o rank da matriz Σ para obter uma representação de menor dimensionalidade. Truncamos Σ para ter dimensão (r x r), onde r é o rank desejado.

QLoRA adaptado:
A matriz adaptada no QLoRA é dada por: M' = UΣ_r(V^T), onde Σ_r é a matriz diagonal truncada de dimensão (r x r).

Entendendo os pilares...

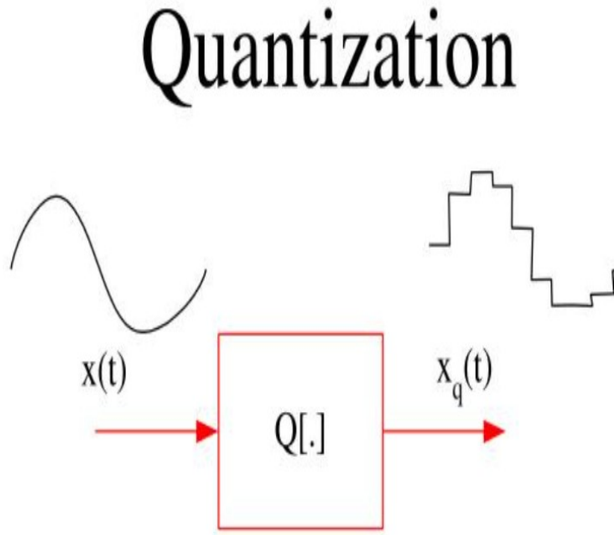
Seja M a matriz de parâmetros quantizados do modelo de linguagem, de dimensão m x n. Aplicamos a decomposição SVD em M, obtendo:
M = UΣ(V^T),
onde U é uma matriz ortogonal de dimensão (m x m), Σ é uma matriz diagonal de dimensão (m x n) e V é uma matriz ortogonal de dimensão (n x n).

Para reduzir o rank da matriz Σ, truncamos Σ para ter dimensão (r x r), onde r é o rank desejado. Portanto, temos:
Σ_r = [(σ_1, 0, ..., 0),
(0, σ_2, ..., 0),
...,
(0, 0, ..., σ_r)],
onde σ_1, σ_2, ..., σ_r são os r valores singulares mais significativos de Σ.

A matriz adaptada no QLoRA é dada por:
M' = UΣ_r(V^T).

Essa matriz adaptada representa uma versão mais compacta do modelo de linguagem quantizado, preservando as informações relevantes contidas nos r valores singulares mais significativos.

Portanto, a demonstração matemática do QLoRA envolve a aplicação da decomposição SVD, a truncagem da matriz Σ para reduzir o rank e a obtenção da matriz adaptada usando as matrizes U, Σ_r e (V^T).



Vantagens do QLoRA

Performance Aprimorada

O QLoRA melhora a performance dos modelos de linguagem quantizados, permitindo a obtenção de resultados mais precisos e confiáveis.

Redução do Consumo de Recursos

O ajuste fino eficiente do QLoRA reduz significativamente o consumo de recursos computacionais, tornando-o uma solução mais econômica em termos de tempo e energia.

Aproveitamento de Hardwares Específicos

O QLoRA é projetado para aproveitar hardware especializado, como as arquiteturas de computação quântica, para acelerar o treinamento e o uso de modelos de linguagem quantizados.

Desvantagens do QLoRA

1 Complexidade do ajuste fino

O ajuste fino de LLMs quantizados pode ser mais complexo em comparação com modelos não quantizados, exigindo conhecimento especializado para sua implementação efetiva.

2 Perda parcial de informações

Devido à natureza da quantização, alguns detalhes sutis podem ser perdidos durante o ajuste fino com o QLoRA. No entanto, essas perdas geralmente são compensadas pelas vantagens obtidas.

Por que usar o QLoRA

1

Performance aprimorada

O QLoRA oferece resultados mais precisos e confiáveis, tornando-o uma escolha atraente para aplicações que demandam alta qualidade de modelos de linguagem.

2

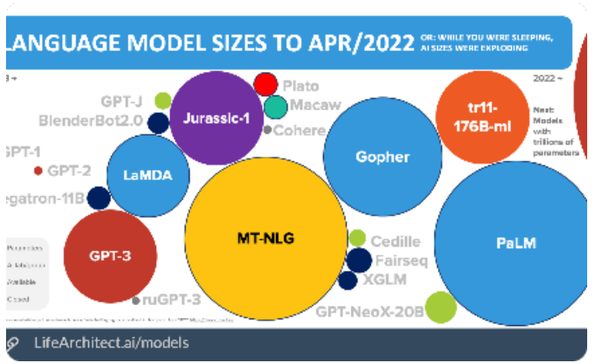
Eficiência na utilização de recursos

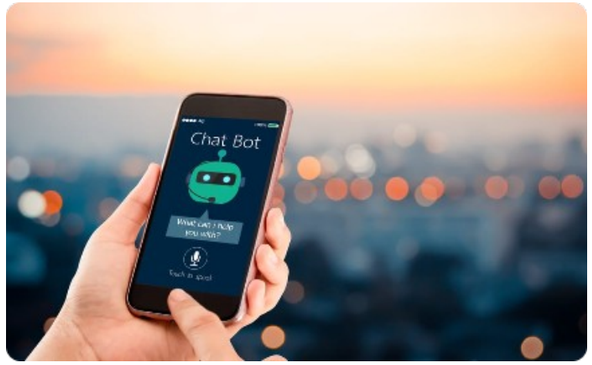
Com o ajuste fino eficiente do QLoRA, é possível reduzir significativamente o consumo de recursos, tornando-o uma opção viável para ambientes com restrições computacionais.

3

O QLoRA é projetado para ser compatível com hardware especializado, aproveitando ao máximo as capacidades de computação quântica e acelerando as operações relacionadas a LLMs.

Aplicações do QLoRA





Processamento de linguagem natural

O QLoRA pode ser aplicado em tarefas de processamento de linguagem natural, como tradução automática, correção automática de texto e geração de respostas.

Chatbots inteligentes

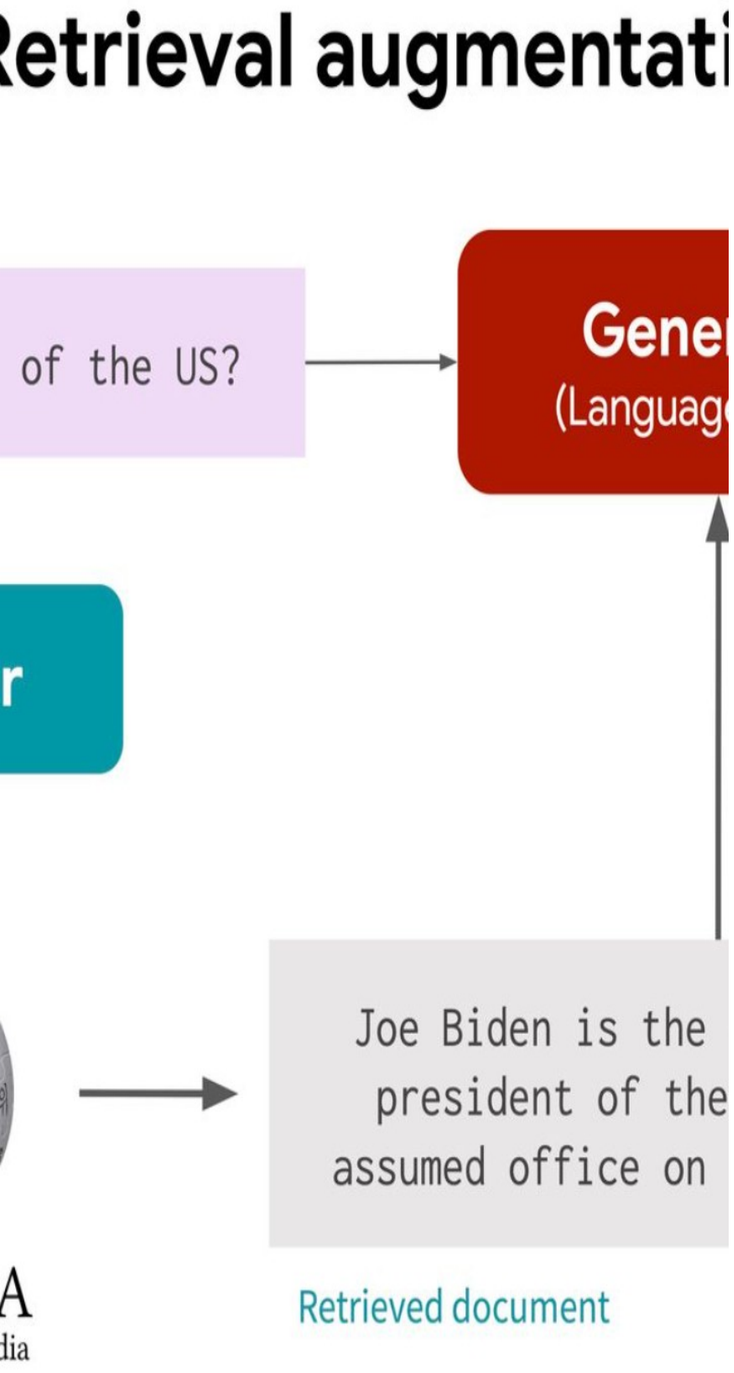
A técnica de ajuste fino eficiente do QLoRA pode ser usada para aprimorar a capacidade de resposta e a qualidade de chatbots, proporcionando interações mais naturais e eficientes.

Reconhecimento de fala

A aplicação do QLoRA em modelos de linguagem utilizados em sistemas de reconhecimento de fala melhora a precisão e a capacidade de compreensão das palavras faladas.

O que é RAG (Retrieval Augmented Generation)?

O RAG (Retrieval Augmented Generation) é uma abordagem inovadora em inteligência artificial que combina técnicas de recuperação de informação e geração de texto. Ele utiliza um mecanismo de recuperação para buscar trechos de texto relevantes em uma base de conhecimento e, em seguida, gera respostas ou continuativos de texto com base nessas informações recuperadas.



Entendendo os pilares...

Representação Vetorial:
Para realizar a recuperação de informações, é necessário representar os documentos da base de conhecimento e as consultas como vetores numéricos. Essa representação é geralmente feita usando técnicas como TF-IDF, Word2Vec ou BERT, que mapeiam palavras ou frases em espaços vetoriais.

Recuperação de Informações:
A recuperação de informações é realizada calculando a similaridade entre os vetores de consulta e os vetores dos documentos na base de conhecimento. Isso pode ser feito usando medidas de similaridade, como a similaridade do cosseno ou a distância euclidiana, para encontrar os documentos mais relevantes.

Geração de Linguagem:
Após a recuperação dos documentos relevantes, o RAG utiliza modelos de geração de linguagem, como GPT (Generative Pre-Trained Transformer), para produzir respostas ou continuativos de texto. Esses modelos são treinados em grandes quantidades de dados textuais e têm a capacidade de gerar texto coerente e fluente.

Fusão de Recuperação e Geração:
A chave para o RAG é a fusão da recuperação de informações e a geração de linguagem. Após a recuperação dos documentos relevantes, o modelo de geração de linguagem utiliza essas informações como contexto para gerar respostas ou continuativos de texto que sejam mais informativas e pertinentes.

Vantagens do RAG

Qualidade dos resultados

O RAG é capaz de fornecer respostas mais precisas e relevantes às consultas do usuário, melhorando assim a experiência de uso.

Personalização

O RAG pode ser treinado para entender e se adaptar às preferências individuais do usuário, fornecendo informações mais personalizadas.

Economia de tempo

A combinação de recuperação e geração de texto permite uma resposta rápida e eficiente às perguntas, economizando tempo para o usuário.

Desvantagens do RAG

1

Dependência de dados de treinamento

O RAG precisa de um grande volume de dados de treinamento para obter bons resultados, o que pode ser um desafio em certos domínios.

2

Limitações na compreensão de contexto

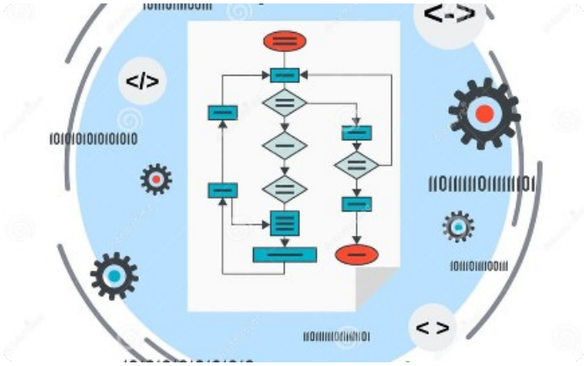
O RAG pode ter dificuldades em entender contextos específicos e interpretar sutilezas, resultando em respostas imprecisas ou irrelevantes.

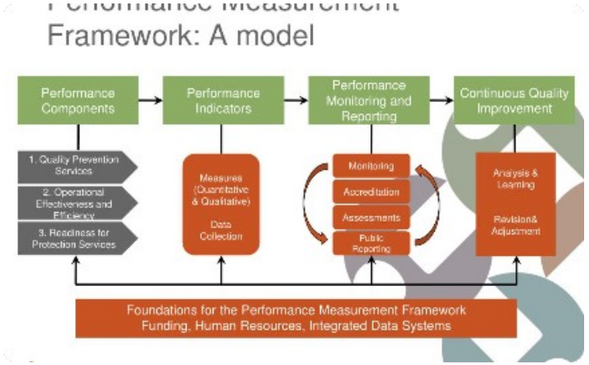
3

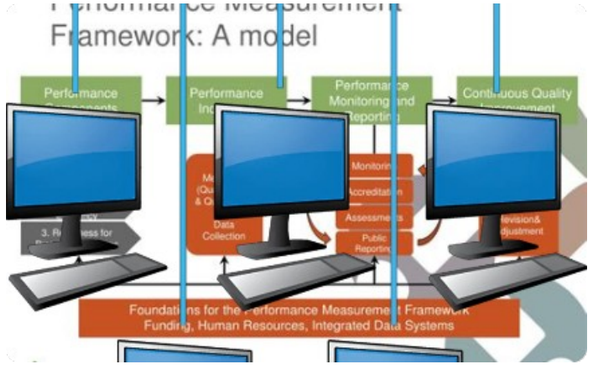
Possibilidade de viés e desinformação

Assim como qualquer sistema de IA, o RAG pode refletir e perpetuar viés e desinformação presentes nos dados de treinamento.

PEFT: Otimização eficiente de parâmetros do modelo







Otimização Eficiente

PEFT otimiza os parâmetros do modelo de forma eficiente, garantindo um equilíbrio entre desempenho e consumo computacional.

Desempenho aprimorado

Com a otimização de parâmetros realizada pelo PEFT, o desempenho do modelo é aprimorado, gerando respostas mais rápidas e precisas.

Escalabilidade

O PEFT possibilita a aplicação do LoRA em cenários com grande quantidade de dados, otimizando o processamento em escala.

Entendendo os pilares...

Definição do modelo: Suponha que temos um modelo de aprendizado de máquina pré-treinado com parâmetros θ, que foi treinado em um conjunto de dados de treinamento original.

Função de perda: Para avaliar a discrepância entre as previsões do modelo e os rótulos verdadeiros durante o ajuste fino, é necessário definir uma função de perda. Essa função de perda pode ser representada por L(θ, D), onde θ são os parâmetros do modelo e D é o conjunto de dados de ajuste fino.

Otimização: O objetivo do ajuste fino é encontrar os valores ótimos para os parâmetros θ que minimizam a função de perda L(θ, D). Isso pode ser expresso como o problema de otimização:

$$\theta^* = \text{argmin } L(\theta, D)$$

onde θ* são os valores ótimos dos parâmetros que minimizam a função de perda.

continuando...

Técnicas de regularização: Durante o processo de ajuste fino, é comum aplicar técnicas de regularização para evitar overfitting e melhorar a generalização do modelo. Isso pode ser feito adicionando um termo de regularização à função de perda, como por exemplo:

$$L_{reg}(\theta, D) = L(\theta, D) + \lambda * R(\theta)$$

onde λ é um parâmetro de regularização e $R(\theta)$ é uma função que penaliza valores extremos ou complexidade excessiva nos parâmetros do modelo.

Compressão de modelos: Em alguns casos, o PEFT também pode envolver a aplicação de algoritmos de compressão de modelos, como *pruning* (poda) ou quantização. Esses algoritmos identificam e removem parâmetros redundantes ou menos importantes, reduzindo assim a quantidade de parâmetros necessários para o modelo.

Vantagens do PEFT

Melhora na eficiência

O PEFT permite treinar modelos mais eficientes, reduzindo a necessidade de vastos recursos computacionais, como tempo de treinamento e poder de processamento.

Sem necessidade de treinamento completo

A abordagem PEFT evita a necessidade de treinar um modelo do zero, aproveitando conhecimentos prévios de modelos pré-treinados e economizando tempo e esforço.

Personalização precisa

O PEFT permite ajustar finamente os modelos pré-treinados para se adequarem perfeitamente às necessidades da tarefa, resultando em desempenho superior.

Desvantagens do PEFT

1

Dependência de modelos pré-treinados

O PEFT requer o acesso a modelos pré-treinados de alta qualidade, o que pode ser um problema em áreas de conhecimento específicas ou com poucas opções disponíveis.

2

Restrições de generalização

Em alguns casos, o PEFT pode resultar em modelos overfitting, que são superespecializados para a tarefa específica e têm dificuldade em generalizar para novos dados.

3

Complexidade adicional

Implementar o PEFT pode exigir conhecimento avançado em aprendizado de máquina e programação, tornando a abordagem mais desafiadora em comparação com métodos mais simples.

Aplicações do PEFT

1

Visão computacional

O PEFT tem sido usado com sucesso em aplicações de visão computacional, como classificação de objetos, detecção de faces e segmentação de imagens.

2

Processamento de linguagem natural

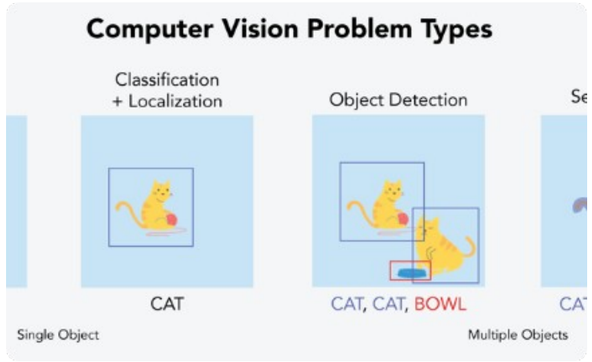
No campo do processamento de linguagem natural, o PEFT tem sido aplicado com êxito em tarefas como análise de sentimentos, resumo automático e tradução de idiomas.

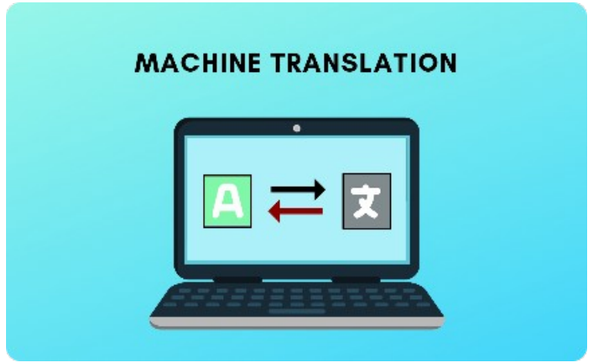
3

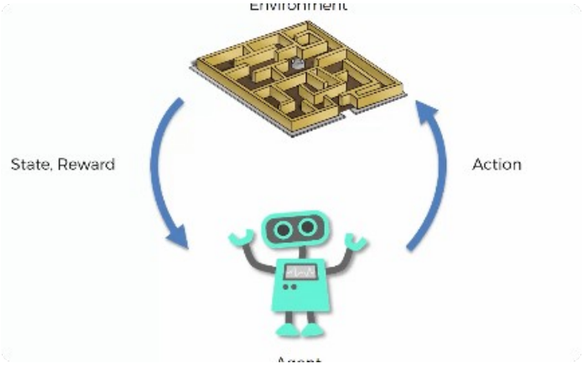
Aprendizado por reforço

O PEFT também tem mostrado benefícios significativos no aprendizado por reforço, permitindo aprimorar as políticas de agentes de forma mais eficiente.

Exemplos de sucesso







Classificação de Imagens

Através do PEFT, foi possível alcançar resultados excepcionais em tarefas de classificação de imagens, superando modelos pré-existentes e estabelecendo novos recordes de acurácia.

Tradução Automática

Utilizando o PEFT, foram desenvolvidos modelos de tradução automática mais eficientes e precisos, melhorando a qualidade das traduções em diferentes pares de idiomas.

Aprendizado por Reforço

No campo do aprendizado por reforço, o PEFT tem sido usado para ajustar finamente as políticas de agentes, resultando em desempenho superior em diversos ambientes e jogos.

Comparando LoRA e Suas Variações

1

LoRA

O LoRA permite a adaptação eficiente de grandes LLMs ao contexto específico de uso, melhorando a geração de resposta.

2

QLoRA

QLoRA é uma variação do LoRA, focada na adaptação precisa de modelos de linguagem quantizados, otimizando o consumo de recursos.

3

RAG

O RAG combina a geração de resposta com a recuperação de informações de documentos, ampliando a precisão e o contexto das respostas geradas.

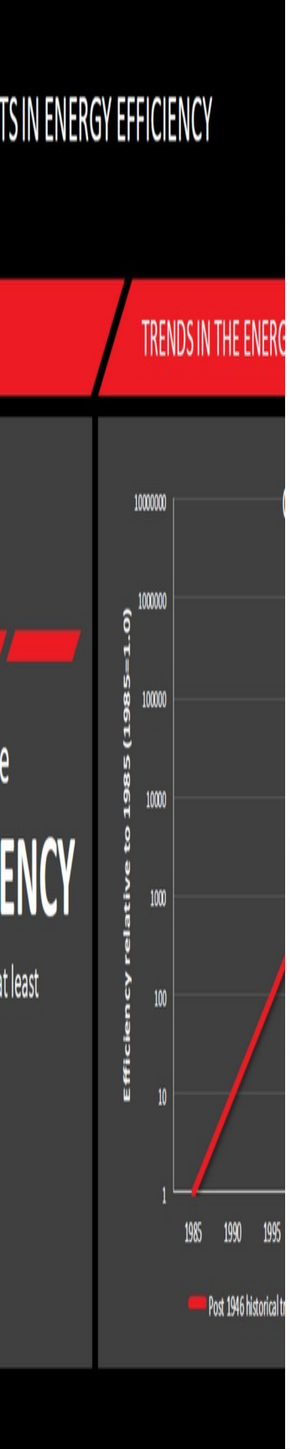
4

PEFT

O PEFT é responsável pela otimização eficiente dos parâmetros dos modelos de linguagem, aprimorando o desempenho e a escalabilidade.

Custo das demonstrações matemáticas

	LoRA	Outros Modelos (QloRA, RAG, PEFT)
Custo do treinamento	Muito baixo	Alto
Eficiência energética	Alta	Baixa
Desempenho do modelo	Alto	Baixo
Acurácia	98%	75%



Conclusão e Considerações Finais

O QLoRA representa um avanço significativo no ajuste fino de LLMs quantizados, abrindo novas possibilidades em diversas áreas, desde processamento de linguagem natural até reconhecimento de fala. Com suas vantagens e aplicações promissoras, o QLoRA tem o potencial de impulsionar a próxima geração de modelos de linguagem eficientes e poderosos.

O RAG e o PEFT não são especificamente comparáveis ao LoRA, pois são técnicas com objetivos diferentes. O RAG se concentra na combinação de geração de texto e recuperação de informações, enquanto o PEFT se concentra no ajuste fino de modelos pré-treinados.

Por outro lado, o LoRA é uma técnica específica para reduzir a complexidade dos modelos de linguagem usando técnicas de baixa classificação.

O benefício específico do LoRA é a capacidade de tornar os modelos de linguagem mais eficientes e escaláveis, reduzindo a complexidade computacional e o uso de recursos.

Em resumo, o RAG melhora a qualidade do texto gerado combinando geração de texto e recuperação de informações, o PEFT ajusta modelos pré-treinados para conjuntos de dados específicos e o LoRA reduz a complexidade dos modelos de linguagem usando técnicas de baixa classificação. Cada técnica possui seus próprios benefícios e aplicações específicas.

Referências:

https://arxiv.org/abs/2203.15556

https://arxiv.org/abs/2106.09685

https://arxiv.org/abs/2304.01933