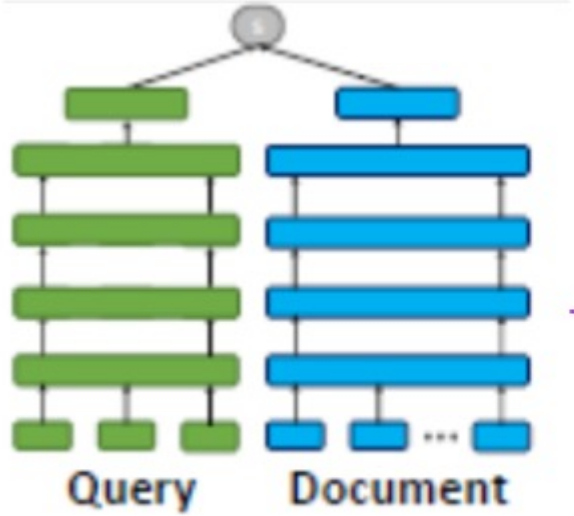
Code Presentation

IA368DD_2023S1: Deep Learning aplicado a Sistemas de Buscas

Student: Marcus Vinícius Borela de Castro

Aula 6/7 - Busca densa (DPR)



(a) Representation-based Similarity

(e.g., DSSM, SNRM)

Arqutitetura dos modelos

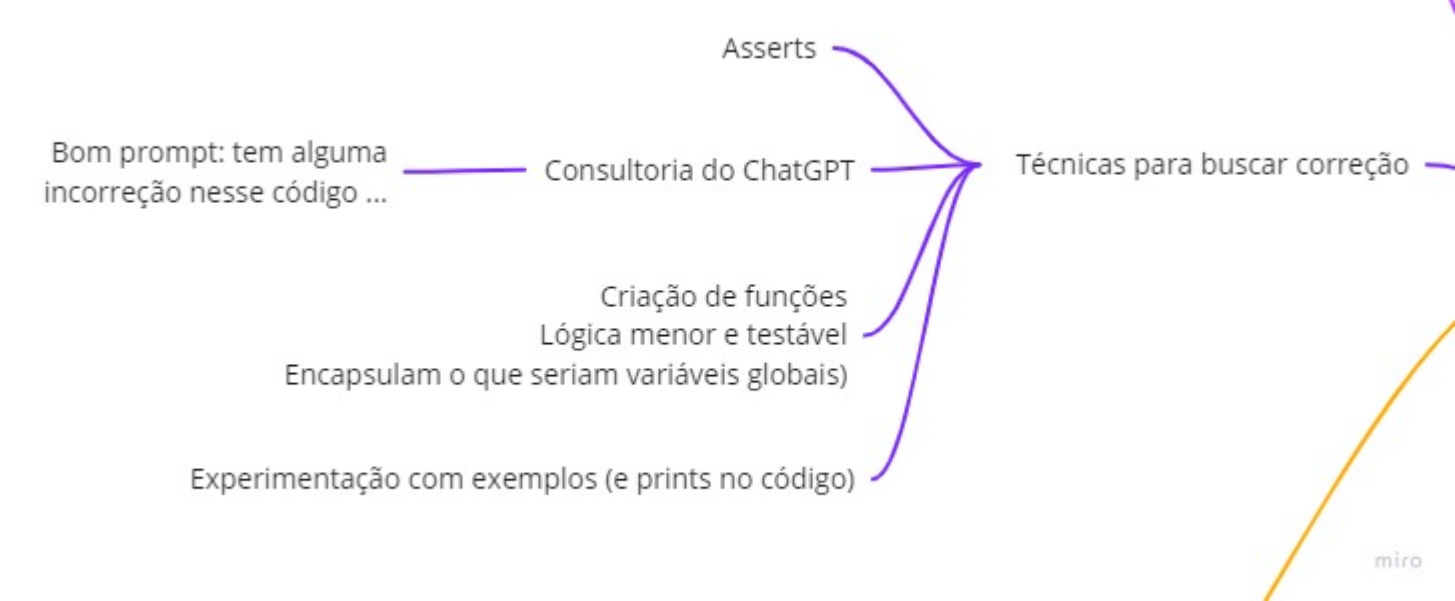
Conceitos

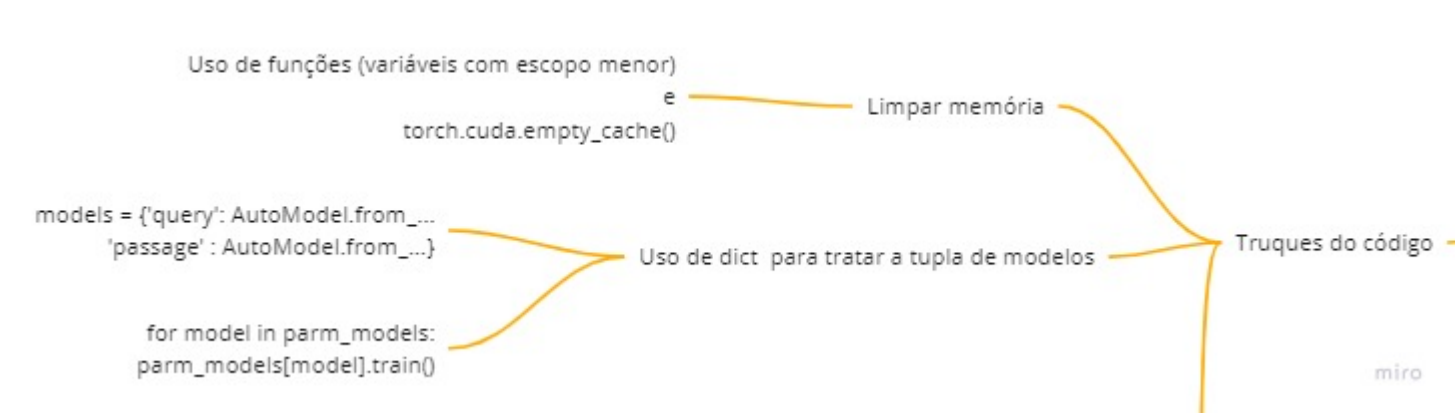
Produto vetorial entre os embeddings dos tokens CLS na última camada

Similaridade

$$L(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-)$$
$$= -\log \frac{e^{\text{sim}(q_i, p_i^+)}}{e^{\text{sim}(q_i, p_i^+)} + \sum_{j=1}^{n} e^{\text{sim}(q_i, p_{i,j}^-)}}$$

Loss (in-batch)


Asserts
Bom prompt: tem alguma incorreção nesse código ...
Consultoria do ChatGPT
Técnicas para buscar correção
Criação de funções
Lógica menor e testável
Encapsulam o que seriam variáveis globais)
Experimentação com exemplos (e prints no código)
miro


Uso de funções (variáveis com escopo menor)
e
torch.cuda.empty_cache()
Limpar memória
models = {'query': AutoModel.from_...
'passage' : AutoModel.from_...}
Uso de dict para tratar a tupla de modelos
for model in parm_models:
parm_models[model].train()
Truques do código
miro

Implementei scheduler linear

com mínimo (reta no início) e máximo

Problemas encontrados

Estava zerando os gradientes apenas antes da chamada do cálculo da loss

após correção: eval loss diminuiu de 0.2 para 0.07

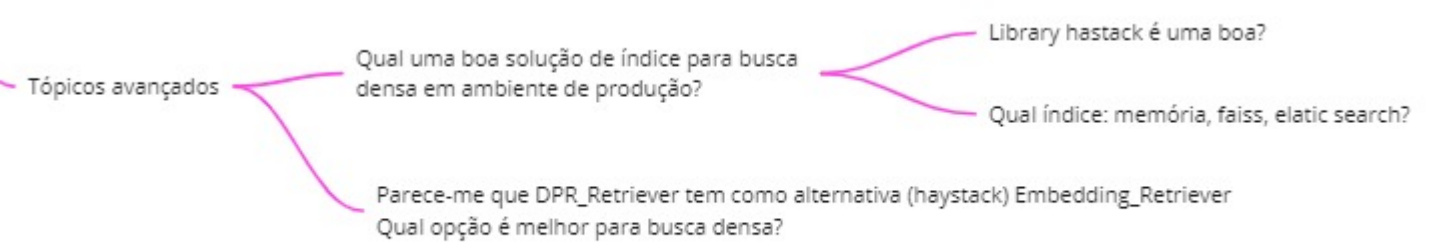
```
for epoch in ...
    for model in parm_models:
        hparam[f'optimizer_{model}'].zero_grad()
    passages_outputs = parm_models['passage'](**batch[0].to(hparam['device']))
    topics_outputs = parm_models['query'](**batch[1].to(hparam['device']))
    for model in parm_models:
        hparam[f'optimizer_{model}'].zero_grad()
    loss = hparam['criterion'](passages_outputs, topics_outputs)
    loss.backward()
    n_examples += len(batch[0]['input_ids']) # Increment of batch size
    for model in parm_models:
      hparam[f'optimizer_{model}'].step()
      hparam[f'scheduler_{model}'].step()
```

Tentei usar biblioteca faiss para indexar mas não deu certo.
Deixei o código ao final do caderno para futuras correções

Resultados interessantes

Valor métrica apurada (nDCG@10) (vetores normalizados, produto escalar, usando cls)
.Modelos em que fiz finetuning: (eval_loss: 0,087)
..busca exaustiva = 36,52
..busca aproximada = 31,23
.Modelo pré-treinado para validar pipeline (all-MiniLM-L12-v2):
..busca exaustiva = 36,97

Dúvidas básicas
Como o cálculo da loss pega valores negativos do próprio batch, podemos concluir que um batch-size maior pode ajudar no desempenho do modelo, já que serão mais amostras?
Precisei implementar linear_warmup_cosine_annealing_lr
Já existe alguma lr_scheduler semelhante em alguma library?


Tópicos avançados
Qual uma boa solução de índice para busca densa em ambiente de produção?
Library hastack é uma boa?
Qual índice: memória, faiss, elatic search?
Parece-me que DPR_Retriever tem como alternativa (haystack) Embedding_Retriever
Qual opção é melhor para busca densa?
miro