

# Notebook – Busca densa

Leandro Carísio

# Conceitos do exercício

1. Cada texto é representado por um vetor
2. Existem 2 encoders de texto que são treinados simultaneamente, um pra documento e outro pra query. O primeiro gera um vetor pro documento ( $v_d$ ) e, o segundo, para a query ( $v_q$ ).
3. O objetivo é que o produto interno entre  $v_d$  e  $v_q$  seja alto quando o doc é relevante para query, e baixo caso contrário.
4. O score é calculado por esse produto interno.

# Problemas e soluções no desenvolvimento

1. No treinamento inicial a loss não estava reduzindo

=> estava calculando a loss do batch somando a loss de cada item. Mudando pra média, reduziu.

2. Na implementação inicial o nDCG@10 estava em 0.03.

=> estava normalizando os vetores da query e do documento antes de calcular a loss. Sem normalizar, resolveu.

=> Em ambos os casos tive ajuda do Eduardo.

# Resultados

Método	nDCG@10
BM25 (Aula 5)	0,5956
BM25 doc. original + expansão (Aula 5)	0,6719
Resultados dessa aula – busca densa	
Implementação – busca em todos os docs	0,3322
Implementação – k means 10 clusters	0,2991
Implementação – treinamento normalizando docs e queries	0,0355
all-mpnet-base-v2	0,5133
all-MiniLM-L12-v2	0,5082
BM25 com 1000 hits e usar busca densa pra reranking	TODO

# Tópico para discussão

O primeiro problema que tive foi resolvido quando parei de normalizar os vetores da query e do documento.

Porque não podemos normalizar os vetores? Não era pra dar o mesmo resultado?

# Obrigado

Leandro Carísio  
carisio@gmail.com