A Novel Relevance Score for Unsupervised Retrieval with Large Language Models

Andrea D'Angelo

Università degli Studi dell'Aquila / Italy

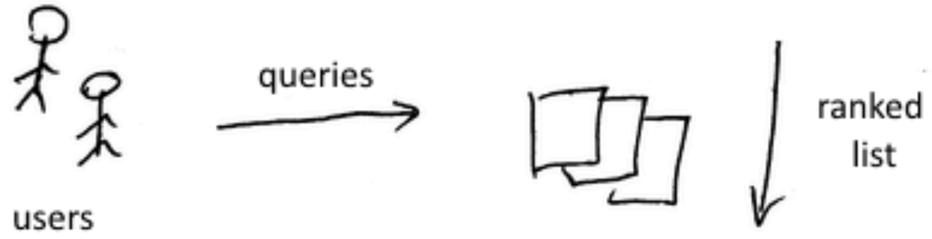
UNIVERSITÀ DEGLI STUDI DELL'AQUILA

DISIM
Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

Context

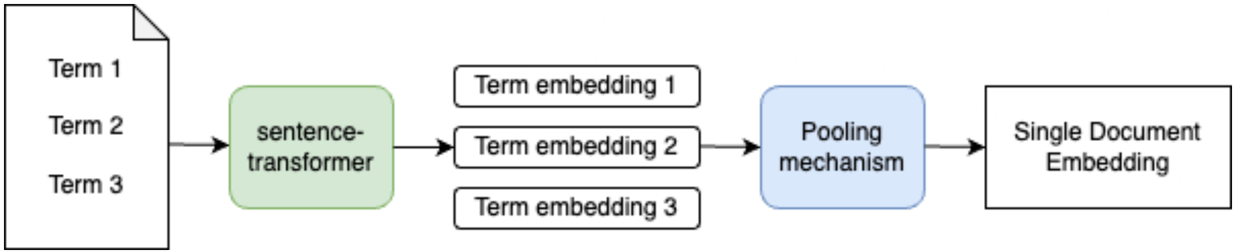
- Document Ranking is the task of returning a ranked list of results given a corpus of documents and a user query.

- Think of a classic Google Search, or a database search, or any Information Retrieval scenario.



Motivation

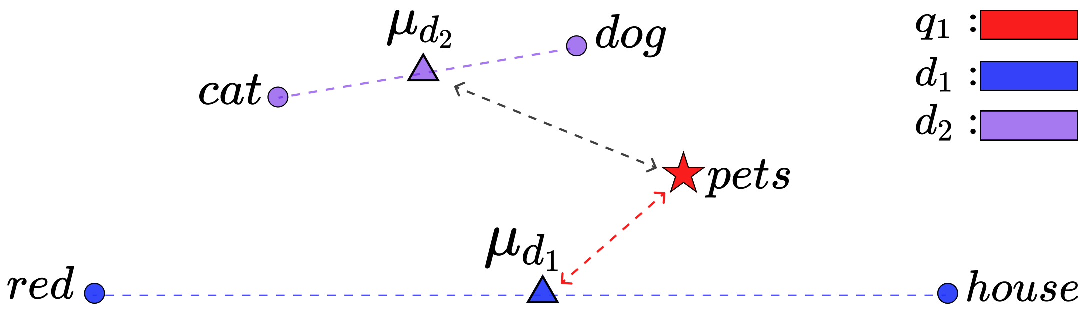
- Most state of the art neural retrievers pool term embeddings into single-vector document representations.



- This approach stems from SBERT [1] and has taken over Document Ranking with DPR [2] and sentence-transformers[3]

Motivation (2)

- Collapsing information into single embedding representations inevitably leads to downfalls.



- Among those, this research highlights the problems of undesirable ranking results and poor explainability.

Motivation (3)

- We will consider the following baselines:

- **MoE (Mean of Embeddings):** the term embeddings are pooled into their mean

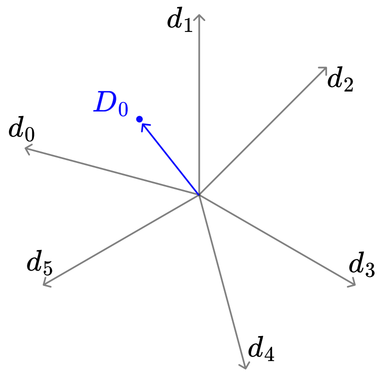
- **XoE (maX of Embeddings):** the term embeddings are pooled into their max

- MoE is the default setting for state-of-the-art sentence transformers.

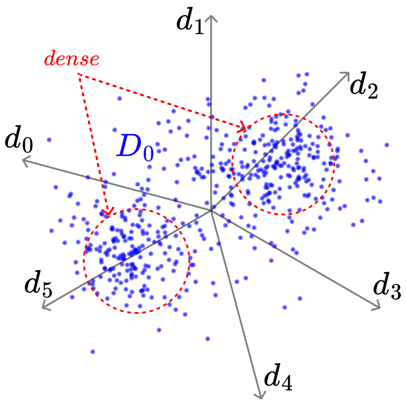
Methodology

- To tackle these problems, we propose **DbU-Cloud**, a novel, density-based Relevance score.

- **DbU-Cloud** does not employ pooling, but rather considers both the **Query's** and the **Document's** sets of term embeddings.



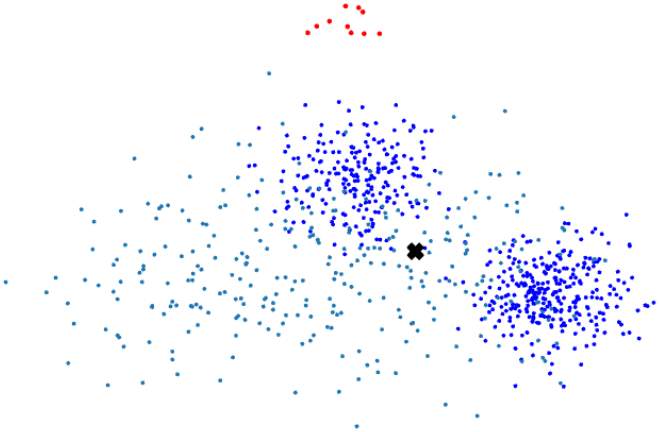
MoE



DbU-Cloud

Methodology (2)

- **DbU-cloud** the relevance score of a set of embeddings with respect to another set of embeddings.



- **DbU-cloud** prioritizes:

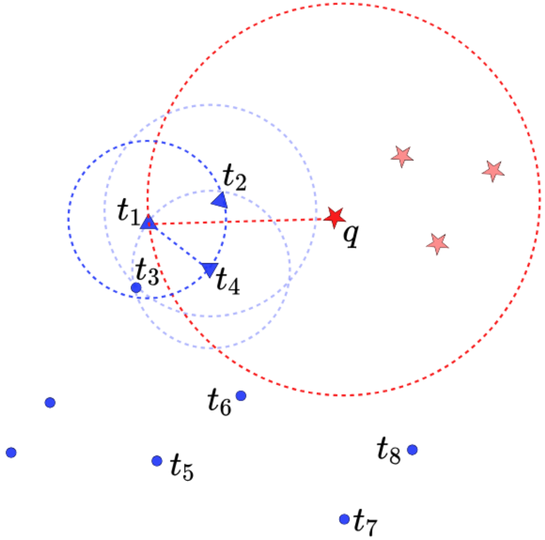
- Density
- Similarity

$$DbU(C_Q, C_D, k) = \sum_{\mathbf{e_q} \in C_Q} \sum_{\mathbf{e_d} \in \mathcal{A}(\mathbf{e_q}, C_D, k)} \frac{\text{local-density}(\mathbf{e_q}, \mathbf{e_d}, C_D, k)}{|C_Q| \cdot |\mathcal{A}(\mathbf{e_q}, C_D, k)|}$$

Methodology (3)

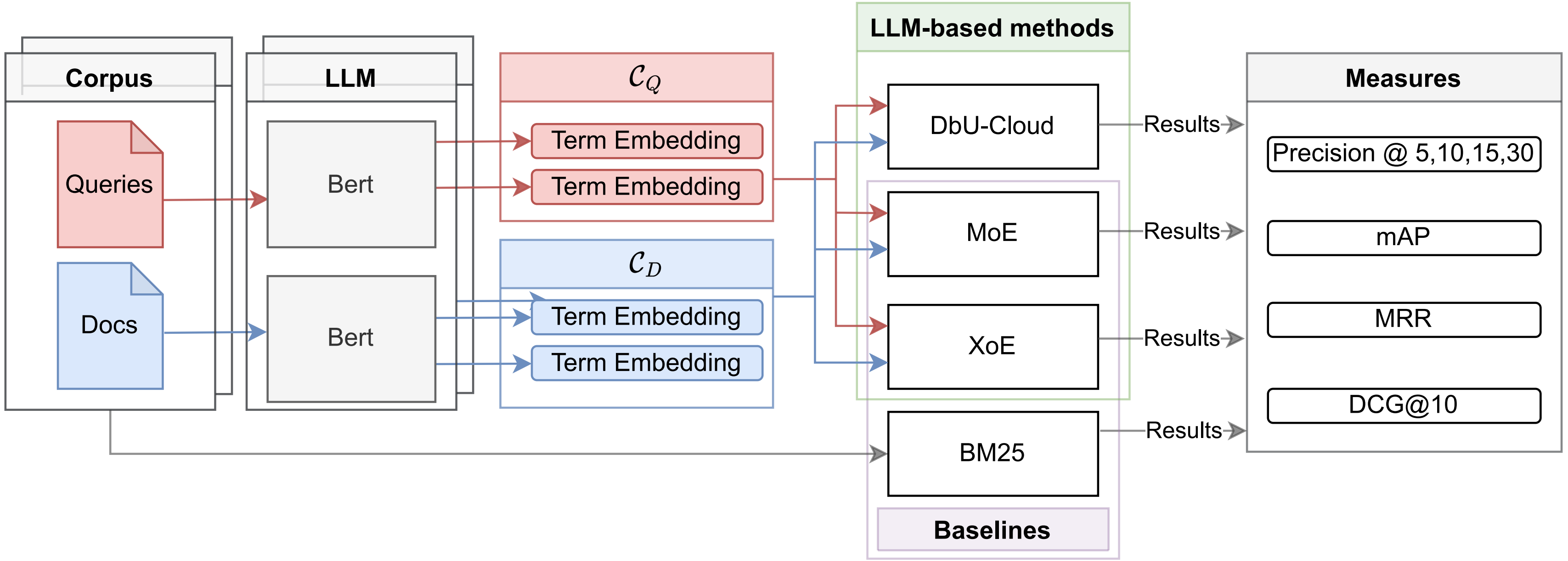
$$DbU(C_Q, C_D, k) = \sum_{\mathbf{e_q} \in C_Q} \sum_{\mathbf{e_d} \in \mathcal{A}(\mathbf{e_q}, C_D, k)} \frac{\text{local-density}(\mathbf{e_q}, \mathbf{e_d}, C_D, k)}{|C_Q| \cdot |\mathcal{A}(\mathbf{e_q}, C_D, k)|}$$

$$\mathcal{A}(\mathbf{e_i}, C_D, k) = \{\mathbf{e_j} \in C_D \setminus \{\mathbf{e_i}\} \mid sim(\mathbf{e_i}, \mathbf{e_j}) \geq akin(\mathbf{e_i}, C_D, k)\}$$



$$\text{local-density}(\mathbf{e_i}, \mathbf{e_j}, C_D, k) = \min\left(\text{sim}(\mathbf{e_i}, \mathbf{e_j}), \text{akin}(\mathbf{e_j}, C_D, k)\right)$$

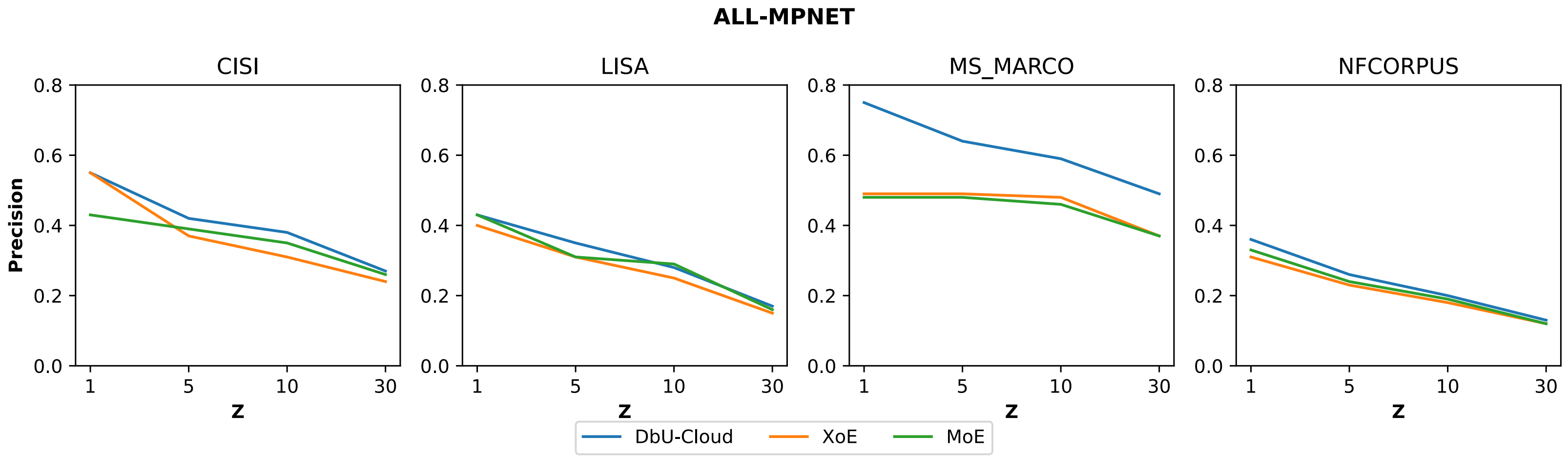
Experimental Settings



- **LLMs:** All-Mpnet-base-v2, DistilRoBERTa, DPR
- **Corpora:** CISI, LISA, MS_MARCO, NFCORPUS

Results

	ALL-MPNET				DistilRoBERTa				DPR			
	MaP	R@10	MRR	DCG@10	MaP	R@10	MRR	DCG@10	MaP	R@10	MRR	DCG@10
XoE	0.160	0.082	0.551	1.400	0.145	0.090	0.549	1.519	0.090	0.071	0.418	1.037
MoE	0.175	0.087	0.550	1.405	0.180	0.100	0.600	1.610	0.110	0.096	0.448	1.115
DbU-Cloud	**0.215**	**0.094**	**0.622**	**1.727**	**0.198**	**0.109**	**0.601**	**1.754**	**0.145**	**0.125**	**0.517**	**1.343**



- DbU-Cloud outperforms MoE and XoE across all LLMs

Main Takeaways

- We identified the main problems caused by the pooling mechanism in Document Ranking.
- We propose a novel Relevance Scoring method, called DbU-Cloud, that removes the pooling mechanism.
- DbU-Cloud promotes density and similarity when computing relevance scores.
- We tested it on multiple corpora and LLMs.
- Results show that DbU-Cloud outperforms pooling across all models.

- 1. SBERT: https://arxiv.org/abs/1908.10084
- 2. DPR: https://arxiv.org/abs/2004.04906
- 3. Sentence-transformers: https://huggingface.co/sentence-transformers