BM 25

10 February 2025 10:13

Formula of BM 25 Score:-

$$\text{score}(D,Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i,D)\cdot(k_1+1)}{f(q_i,D)+k_1\cdot\left(1-b+b\cdot\frac{|D|}{\text{avgdl}}\right)}$$

$$\text{IDF}(q_i) = \ln\left(\frac{N-n(q_i)+0.5}{n(q_i)+0.5}+1\right)$$

$$= \ln\left(\frac{N-n(q_i)+0.5+n(q_i)+0.5}{n(q_i)+0.5}\right)$$

$$= \ln\left(\frac{N+1}{n(q_i)+0.5}\right)$$

Nomenclature:-

- score (D, Q) = Score of a doc | query
- $f(q_i, D)$ = No of times q_i occurs in doc D
- |D| = Length of doc D [no of words in D]
- avgdl = Avg length of docs

Formula of TF-IDF Score:-

$$\text{tf}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$$ → Popularity of t in D

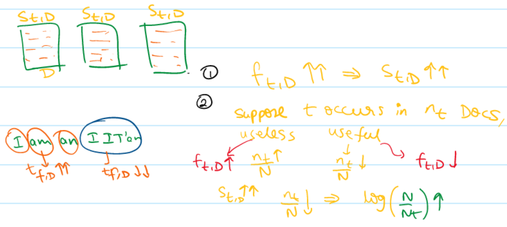
$$\text{idf}(t,D) = \log \frac{N}{|\{d : d \in D \text{ and } t \in d\}|}$$ → Uniqueness of t in all Docs

$$\text{Score}(D,Q) = \sum_{t \in Q} tf(t,D) \times idf(t,D)$$

Nomenclature:-

- score (D, Q) = Score of a doc | query
- $f_{t,D}$ = No of times t occurs in doc D
- $tf(t,D)$ = (no of times t occurs in D) / (Total no of words in D)

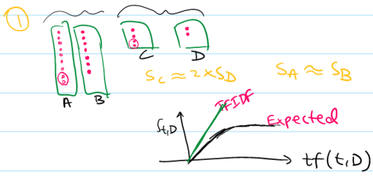
Intuitive explanation of the formula



$$\text{score}(D,Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i,D)\cdot(k_1+1)}{f(q_i,D)+k_1\cdot\left(1-b+b\cdot\frac{|D|}{\text{avgdl}}\right)}$$

$$\text{IDF}(q_i) = \ln\left(\frac{N-n(q_i)+0.5}{n(q_i)+0.5}+1\right)$$

Problems with TF-IDF

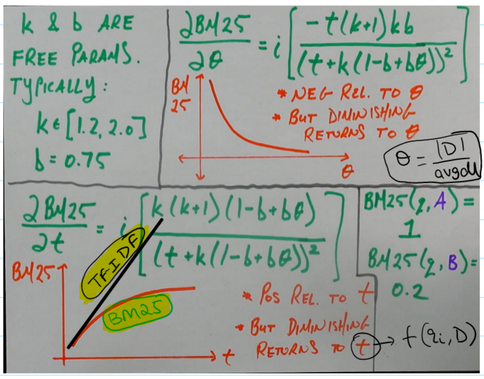


② Doc Length bias:-

Longer Doc → more chances of finding t
⇒ tf ↑↑
⇒ Score ↑↑↑

③ Lack of flexibility in TF IDF:-

k_1 & b → Hyper parameters in BM25



BM25 (Best Matching 25) is a widely used ranking algorithm in information retrieval systems. It is employed in various applications and platforms:

Search Engines

- Web search engines like Google, Bing, and Yahoo use BM25 or similar algorithms to determine the relevance of search results3.
- Enterprise search systems in large organizations utilize BM25 to provide employees with relevant documents and information from internal databases3.

E-commerce

- Online shopping platforms often implement BM25 or similar algorithms to rank products based on relevance to user queries and provide personalized product recommendations3.

Information Retrieval Systems

- BM25 serves as a strong baseline in information retrieval research, particularly in the TREC Web track5.
- It is used in document retrieval systems to rank documents based on their relevance to search queries1.

Vector Databases

- Vector databases like Milvus integrate BM25 to enhance search relevance and efficiency1.

Hybrid Search Systems

- Many real-world search applications combine BM25-based search with vector-based semantic search powered by large language models (LLMs)4.

AI and Machine Learning

- BM25 is often integrated with LLMs using Retrieval-Augmented Generation (RAG) to improve search and retrieval performance4.
- It can be used as a cost-effective semantic cache when integrating LLMs into production systems4.

Specific Platforms

- Azure AI Search uses BM25 as its default relevance scoring algorithm11.
- Elasticsearch implements BM25 as its default similarity ranking algorithm10.
- SAP HANA Cloud Database includes BM25 search functionality9.
- LangChain and Weaviate, popular AI development frameworks, offer BM25 retrieval options1213.

By leveraging BM25 in these diverse applications, developers and researchers can create more effective and efficient information retrieval systems across various domains