Information Retrieval

What is IR

- The process of obtaining relevant information from a large repository (e.g., documents, web pages, databases).
- **Documents**: Textual data (web pages, articles, books)
  - Multimedia & Maps
- **Queries**: User input expressing information need.
- **Indexing**: Creating efficient data structures for fast retrieval.
- **Ranking**: Ordering results by relevance.

Applications

- Search engines
- Document retrieval
- Recommender system
- …

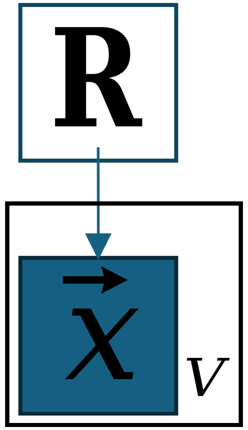
BM25

- probabilistic retrieval model based on the Bag-of-Words assumption
- Probability Ranking Principle
    - : document
    - : relevance; {1, 0}
- Interested only in ranking

Binary Independence Model (BIM)

- Given query

Naïve Bayes



- : binary representation of document (Term-existence)
- Naive Bayes conditional independence assumption: presence/absence of a word in a document is independent of the presence/absence of any other word

Binary Independence Model (BIM)

- Assumption:
  - Terms not in query does not impact relevance

Retrieval Status Value

- Retrieval Status Value
- If assume RSVIDF

BM25

- Best match 25
- Words are drawn independently from the vocabulary using a multinomial distribution
- Distribution of term frequencies (tf) follows a binomial distribution – approximated by a Poisson
- Assume that term frequencies in a document follow a Poisson distribution

Poisson Distribution

- Models the probability of the number of events occurring in a fixed interval of time/space, with known average rate
- Examples
  - Number of cars arriving at the toll booth per minute
  - Number of typos on a page
- Also be used to approximate binomial
- Assume that term frequencies in a document follow a Poisson distribution
  - Implies fixed document length
  - Reasonable fit for “general” words, but poor for the topic-specific words

Extensions

- Term is either regular or topic related
- Extend the Poisson as mixture of Poisson
  - Use a simple function to approximate:
  - controls term frequency scaling
- Document length normalization
  - b: a parameter between 0 and 1. 0 means no length normalization

BM25

- Normalize term frequency using document length

- Ranking function

Google

- Google was founded on September 4, 1998, by Larry Page and Sergey Brin.
- Google began in January 1996 as a research project by Larry Page and Sergey Brin while they were both PhD students at Stanford

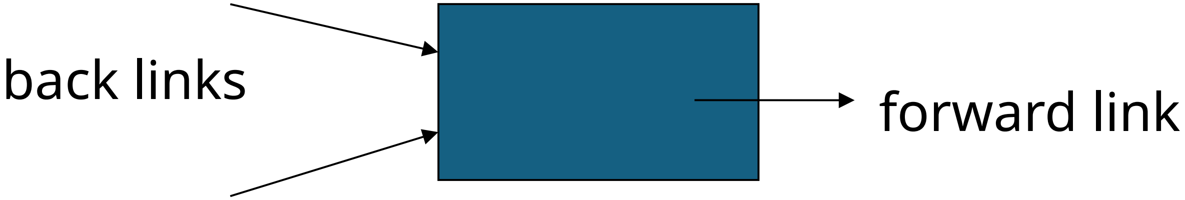
- Brin, Sergey, and Lawrence Page. "The anatomy of a large-scale hypertextual web search engine." *Computer networks and ISDN systems* 30.1-7 (1998): 107-117.

Ranking

- Webpage corpus is not a controlled collection
    - BM25/tf-idf only consider relevance
    - Reputation?
- Hit rate
    - Rank higher if it is visited more frequently
    - Fake hits
    - Cold start for new pages
- Citation
    - A paper is important if it is cited by many papers
    - Not well-controlled

PageRank

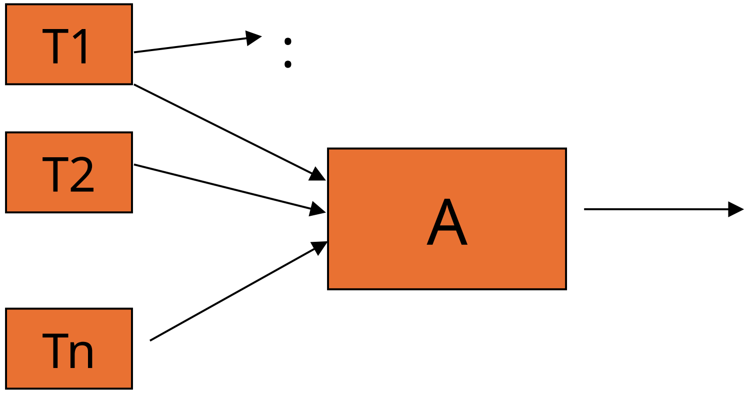
- A page with many links to it is more likely to be useful than one with few links to it
  - Just like citation
- The links from a page that itself is the target of many links are likely to be particularly important
  - This is something new



Each link has different weight

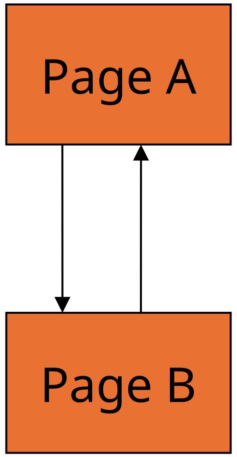
PageRank

- Each page is ranked using a value called PageRank (PR)
- A page’s PR depends on the PRs of its back link pages
    - : damping factor, normally this is set to 0.85
    - : pages linking to page A
    - : PageRank of page A
    - : PageRank of page pointing to page A
    - : the number of links going out of page



PageRank Example

- Assign each page an initial rank val
  - Could be any number (seed)
- Repeat calculations until converge



Seed = 0
1)
PR(A)= 0.15 + 0.85 * 0 = 0.15
PR(B)= 0.15 + 0.85 * 0.15 = 0.2775
2)
PR(A)= 0.15 + 0.85 * 0.2775 = 0.385875
PR(B)= 0.15 + 0.85 * 0.385875 = 0.47799375
3)
PR(A)= 0.15 + 0.85 * 0.47799375 = 0.5562946875
PR(B)= 0.15 + 0.85 * 0.5562946875 = 0.622850484375

Seed = 40
1)
PR(A)= 0.15 + 0.85 * 40 = 34.25
PR(B)= 0.15 + 0.85 * 0.385875 = 29.1775
2)
PR(A)= 0.15 + 0.85 * 29.1775 = 24.950875
PR(B)= 0.15 + 0.85 * 24.950875 = 21.35824375
3) ……

d= 0.85
PR(A)= (1 – d) + d(PR(B)/1)
PR(B)= (1 – d) + d(PR(A)/1)

Supervised Ranking Methods

- Binary classification

- Learning to rank
  - Pointwise (regression problem)
    - : predicting the real-value or ordinal score of x
    -
  - **Pairwise**
    - : classification problem for a given pair
    - Usually implemented with a scoring function:

  - Listwise

Training loss

- max-margin loss

ranking score

- Cross-entropy loss

- Negative log-likelihood loss

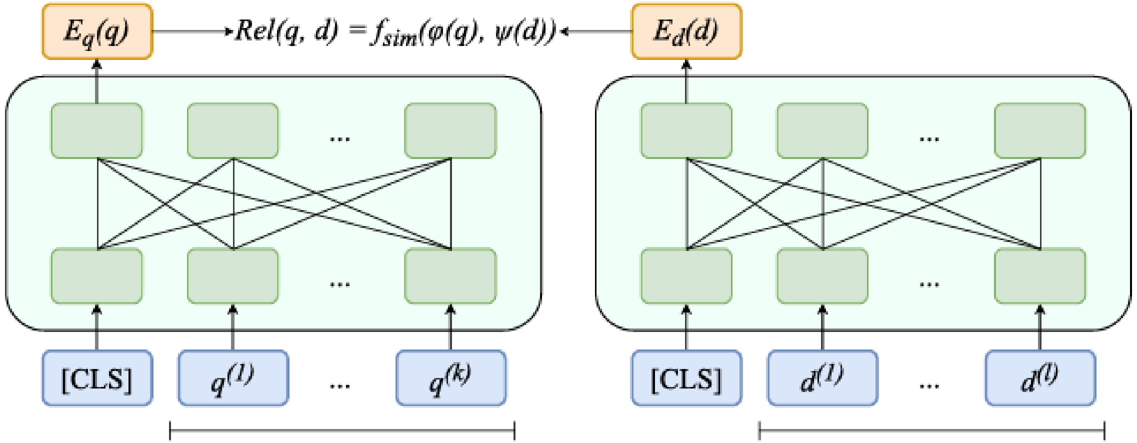
- : small set of negative samples for query

Dense Retrieval

- Sparse
  - bag-of-words
  - Lexical similarity
- Learning to rank
  - Feature-based
  - hand-crafted features or embedding (pre-PLM)
- Dense
  - Pretrained Language Model (PLM) -based
  - Semantic similarity

Zhao, Wayne Xin, et al. "Dense text retrieval based on pretrained language models: A survey." *ACM Transactions on Information Systems* 42.4 (2024): 1-60.

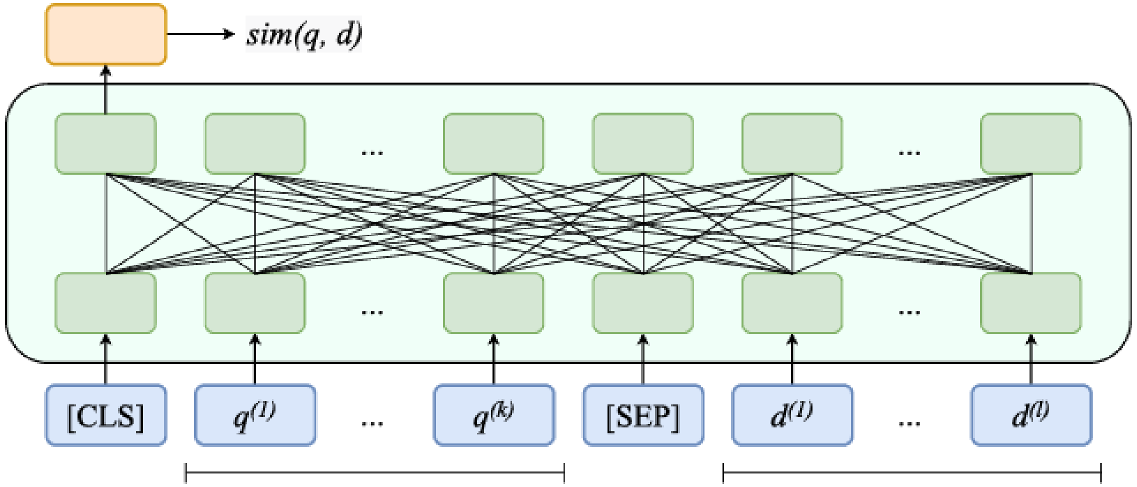
Encoder Architecture



query

documen

Bi-encoder architecture



query

documen

cross-encoder architecture

- Can you think about potential pros and cons them?
- effectiveness and efficiency trade-off

Training – negative selection

- In-batch Negatives
  - given a query, the positive texts paired with the rest queries from the same batch are considered as negatives.
  - Assume that there are queries () in a batch
  - How many in-batch negatives for each query?
- Cross-batch Negatives
  - multi-GPU setting
  - Assume there are GPUs, queries () in a batch in one GPU
  - How many in-batch negatives for each query?

Training – negative selection

- Hard Negatives
    - irrelevant texts but having a high semantic similarity with the query
- static hard negatives
    - sample negatives from top retrieval results from some other retrievers, such as BM25
    - first clusters the queries before training and then samples queries out of one cluster per batch
- dynamic hard negatives
    - sample from the top retrieved texts by the optimized retriever itself as negatives
- Denoised hard negatives: reduce false negatives

Zero-shot retrieval

- Shot
    - Examples
    - Few-shot: few examples
    - Zero-shot: no examples
- Zero-shot
    - Instruction only. E.g., prompt
    - Train in domain A, apply in domain B
    - Train a retriever for Wikipedia, apply it in PubMed

Evaluation

- Recall@k
  - Percentage of queries that the relevant documents are ranked within top k

- NDCG (Normalized Discounted Cumulative Gain)
  - Considers the position of a relevant text (higher the better)
  - https://www.evidentlyai.com/ranking-metrics/ndcg-metric
- MRR (Mean Reciprocal Rank)
  - : the rank of the first retrieved positive text for query q

Recommender System

- eCommerce
- Job matching
- News feed
- …

- For a user, return a ranked list based on preference

Recommender System

- Features of users
  - History of rating/click
  - Social network
- Features of items
  - Topic labels (genre)
  - Reviews

- Content Filtering
- Collaborative Filtering

Content Filtering

- description of the item
- a profile of the user's preferences
  - A model of the user's preference.
  - A history of the user's interaction with the recommender system
- estimate the probability that the user is going to like the item

Collaborative Filtering

- personal tastes are correlated
  - If Alice and Bob both like X and Alice likes Y then Bob is more likely to like Y

Items / Users	Movie 1	Movie 2	Movie 3	Movie 4
Alex	1		5	4
George	2	3	4	
Mark	4	5		2
Peter			4	5

Collaborative Filtering

- Matrix factorization
  - This matrix is large and very sparse
  - Non-negative Matrix Factorization
  - Low dimensional representation of user and items
  - Embed users and items in the same space
- Hybrid approach

Evaluation

- Fixed test data
  - Netflix challenge: predict users’ ratings, MSE
- User study
  - Small scale
- A/B test (online)
  - Click through rate
- Diversity, novelty, and others