IR Pipelines and Modern IR

COMP3009J: Information Retrieval

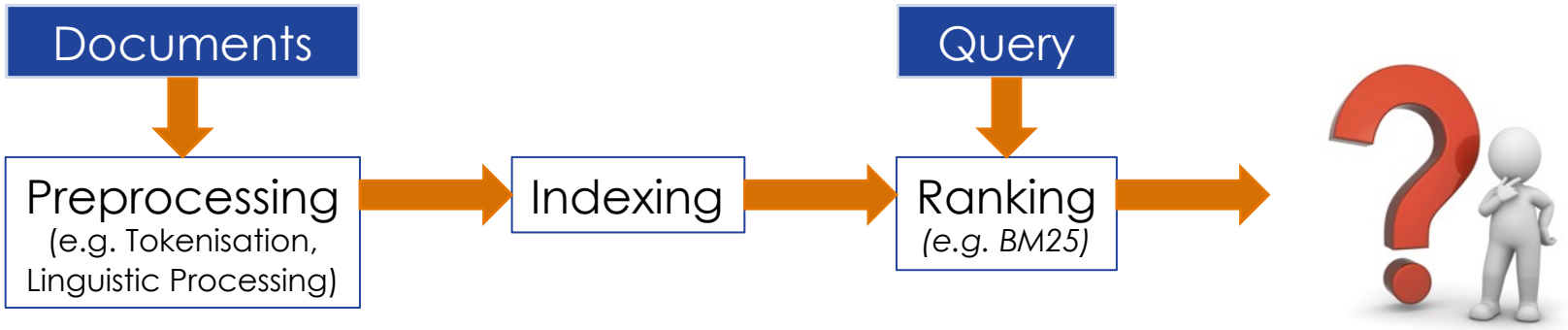
Dr. David Lillis (david.lillis@ucd.ie)

UCD School of Computer Science
Beijing Dublin International College

Recap: where are we now?

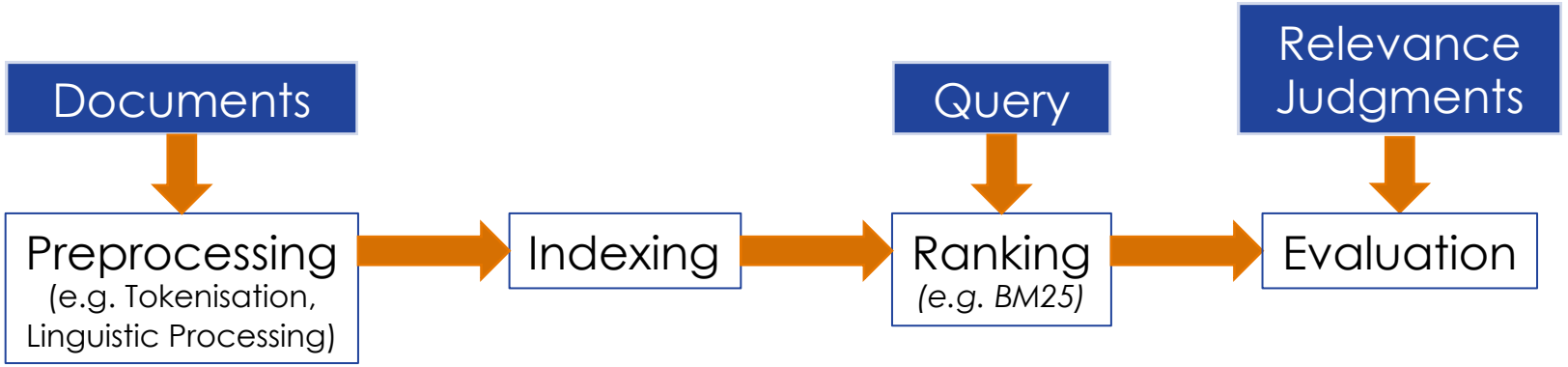
- So far we have looked at **classic** Information Retrieval, much of which is still relevant today. We know…
  - What IR is.
  - How an index can be built.
  - How several models can perform retrieval.
  - How evaluation can be done.
- Thus all the basic components are in place.
- Next we will look at how this process can be improved.

Classical IR Pipeline



1. Documents are **preprocessed** to prepare them for use in the IR system.
2. The **indexer** creates a suitable data structure so that retrieval can happen.
3. A **ranking** method matches queries against the index to produce a **ranked list of results** for the user (or to an evaluation step).

Classical IR Pipeline (Experimental)

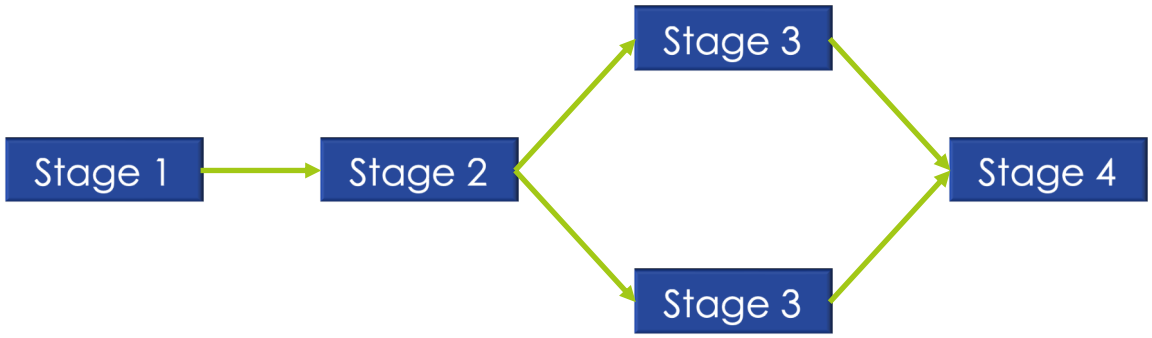


- Alternative pipeline with Evaluation component.
  - Used in experimentation – to evaluate the performance of the system that has been designed.
    - (this is evaluation in terms of the system's **effectiveness**)

Aside: Pipeline Design Pattern

- We call this design pattern a **pipeline** because the output of one process becomes the input of the next stage in the process.
- This is similar to a **linear pipeline**, where each stage happens sequentially.

- An alternative is a **non-linear** pipeline, where some stages can happen in parallel.



Issues with Classical Pipeline (so far)

- **Vocabulary mismatch** is a problem. The words chosen by the user may not match the words used in a relevant document.
  - ***polysemy***: the same word can have different meanings (e.g. "bank").
  - ***synonymy***: two different words can have the same meaning (e.g. plane/airplane/aeroplane/aircraft).
- How can we be confident that our choice of **retrieval model** is the best for each query?
  - Could we use a combination of ranking models instead of just one?

Issues with Classical Pipeline (so far)

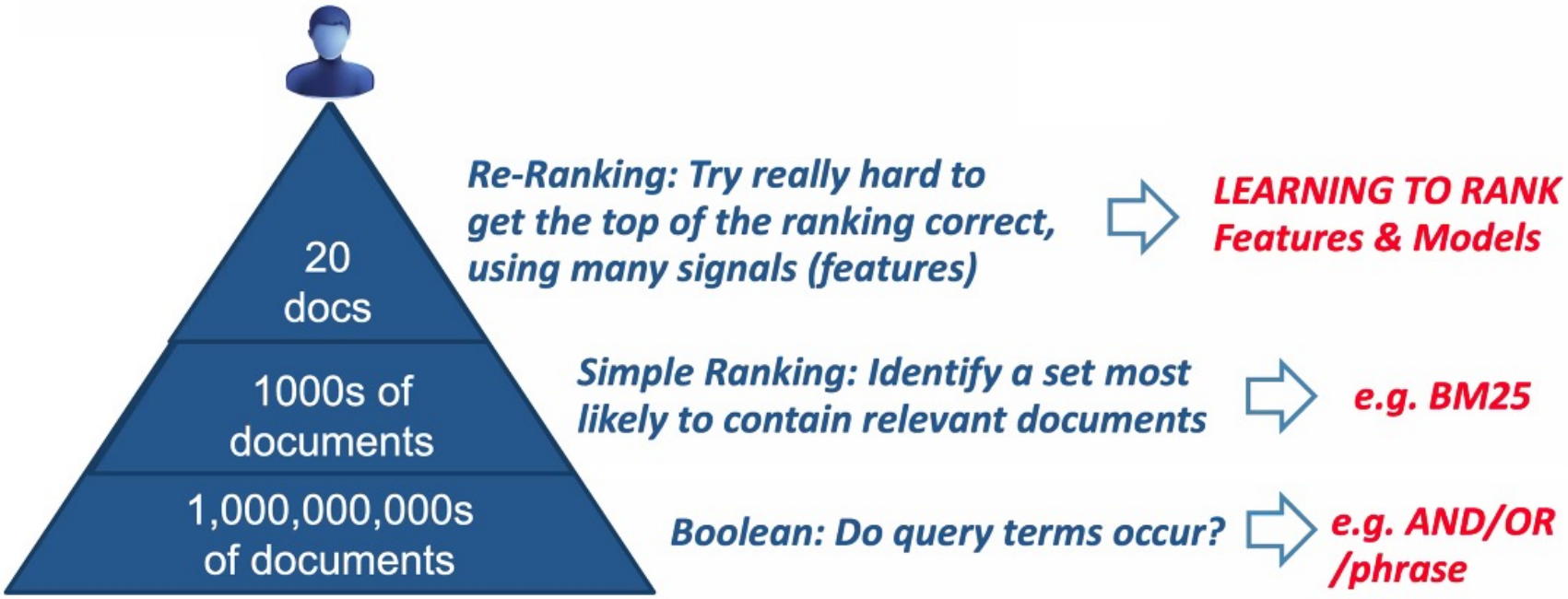
- Only the **document contents** are taken into account.
  - Particularly on the web, there is a lot more information available.
    - The **link structure** of the web:
      - Where does this page link to?
      - What other pages link to this page?
      - What text do other pages use in their links to this page (in their **anchor text**)?
    - The **URL** may contain useful information?
    - For a large search engines, they have **query logs** to record previous user's behaviour when they give the same query.

Issues with Classical Pipeline

- Doesn't make use of recent advances in **machine learning** (especially **deep learning**) and **Natural Language Processing** (NLP) and **Natural Language Understanding** (NLU).
  - Machine Learning has transformed most data-processing and data-analysis tasks.
  - Large Language Models (LLMs) such as **BERT** (Google, 2018), **T5** (Google, 2020), **GPT-4** (OpenAI, 2023) have made huge progress in a variety of NLP, NLU and Machine Translation tasks.
  - **BUT…** Languages models are *slooooooow*…

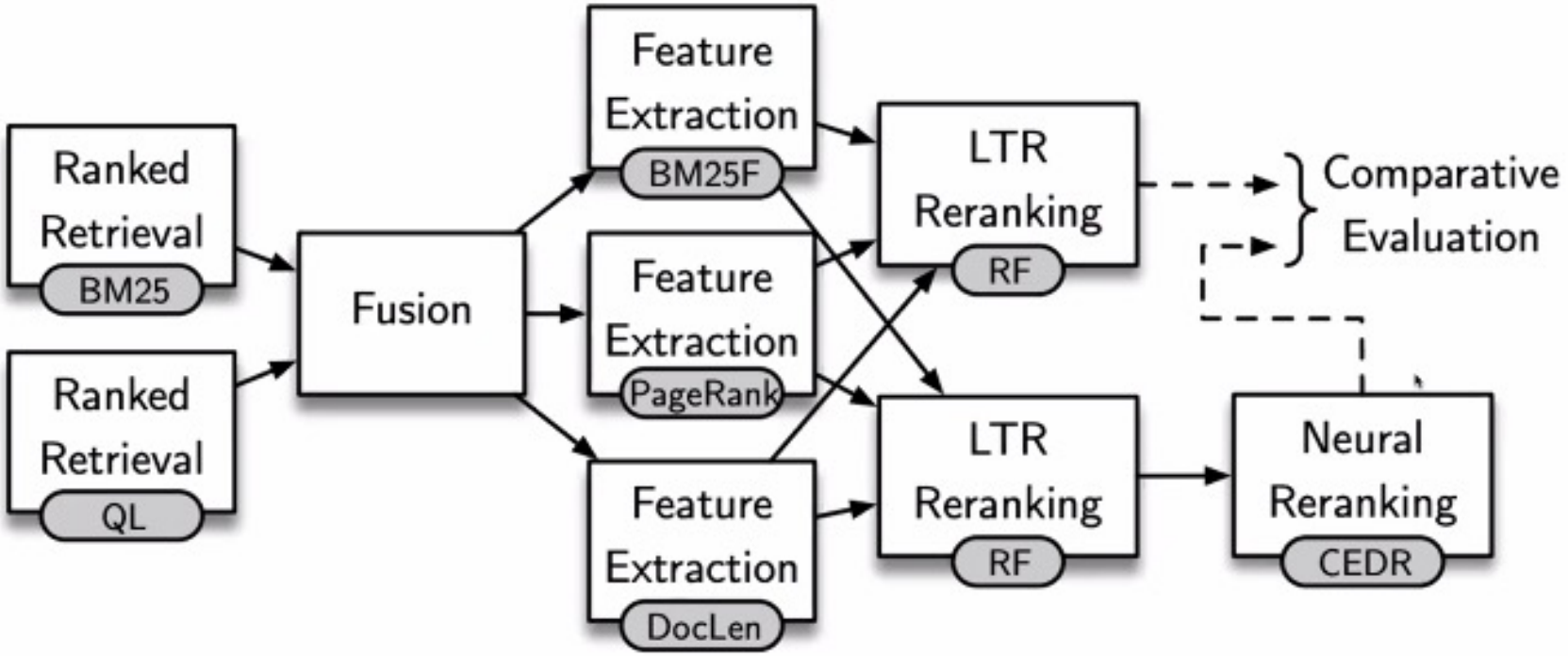
Speed is of the essence…

- **Speed is essential.** According to Microsoft:
  - *"Even a **100ms** latency has been shown to invoke negative user reactions"*[1]



[1] Rosset, C., Jose, D., Ghosh, G., Mitra, B., & Tiwary, S. (2018). Optimizing Query Evaluations Using Reinforcement Learning for Web Search. *SIGIR 2018*. https://doi.org/10.1145/3209978.3210127

A More Complex IR Pipeline



Source: MacAvaney, S., McDonald, C., and Tonollotto, N., IR From Bag-of-words to BERT and Beyond through Practical Experiments, ECIR 2021.

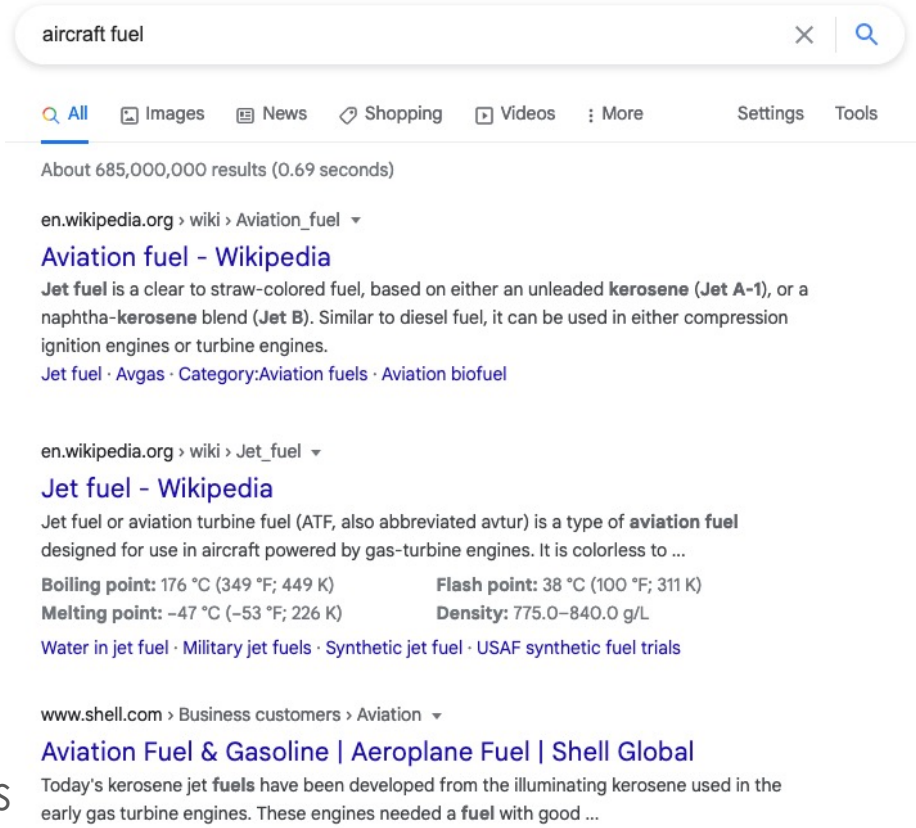
Pseudo-Relevance Feedback

- **Relevance Feedback** is when users are shown a set of documents in response to a query, and tell the system which ones are relevant.
  - The system can then use this information to re-run the search(with different term weights) and hopefully find more relevant documents that are similar to the ones the user has said are relevant.
  - **However**, users don't like giving relevance feedback!
- **Pseudo-Relevance Feedback** simulates relevance feedback by *assuming* that the top *k* ranked documents are relevant, and then re-running the search in the same way.
  - **Why does this work?** *"[T]erms that occur consistently in relevant documents also tend to appear consistently in top ranked documents."*[1]
  - **But** it's not perfect: danger of **query drift** towards the topic of the top-ranked documents.

[1] Zhao, L., and Callan, J., Term Necessity Prediction, CIKM 2010. https://doi.org/10.1145/1871437.1871474

Query Expansion

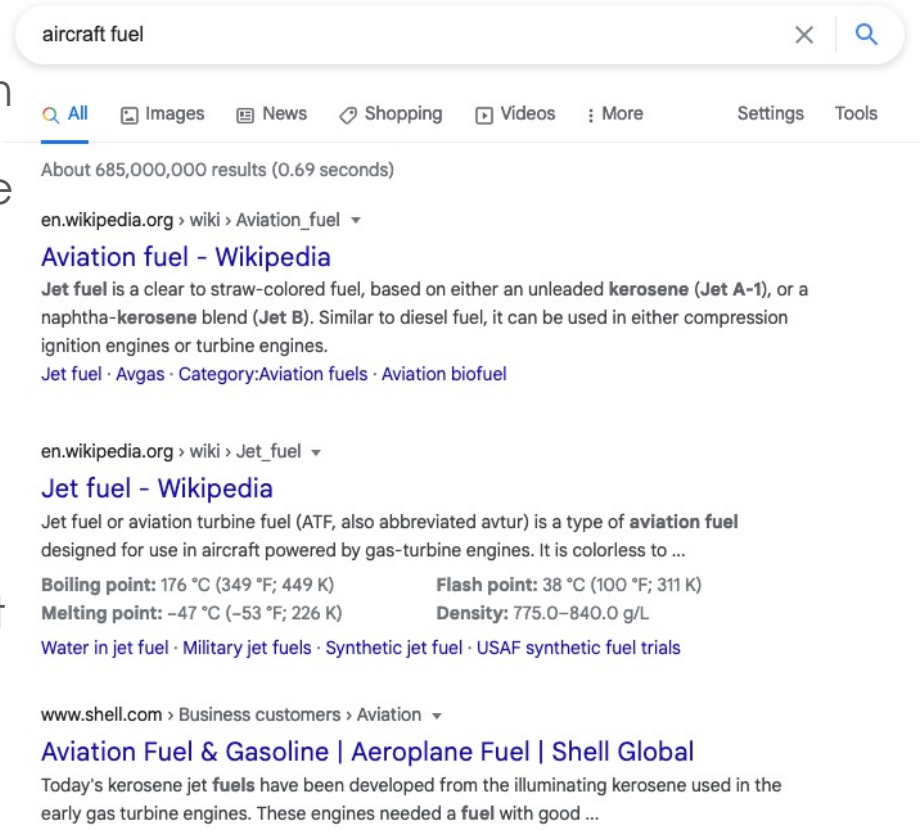
- Related terms are **added to the query** to increase the chance of matching relevant documents.
- Many sources of related terms:
  - A **manually-created thesaurus**, such as WordNet[1].
  - An **automatically-created thesaurus** from some external corpus like a web crawl, or Wikipedia[2].
  - **Word embeddings**, where words are represented by vectors (e.g. word2vec, GloVe, ELMo and BERT-based embeddings).



[1] https://wordnet.princeton.edu/
[2] For example, through DBPedia (https://wiki.dbpedia.org/)

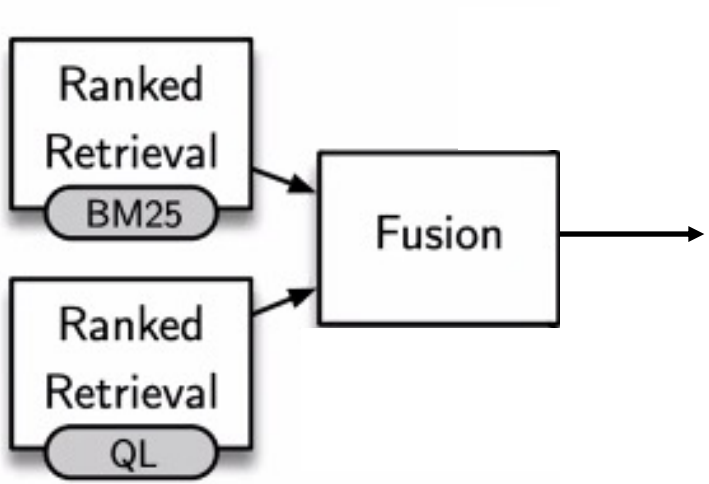
Query Expansion

- Many sources of related terms:
  - **Query Log** mining, where a search engine looks at the behaviour of previous users who have given the same query.
  - **Target-corpus based techniques**, based on the document collection being searched. Two categories:
    - *Distribution-based:* compare the distribution (frequency) of terms in the (pseudo-) relevant documents compared to the whole corpus.
    - *Association-based:* select terms based on their association (co-occurrence) with the query terms.



Fusion

- **Input:** a set of results (for one query) from several Ranked Retrieval systems.
- **Output:** a single, combined set of results that are (hopefully) better than the input results in isolation.
- Can be based on:
  - the **scores** calculated by the Ranked Retrieval systems;
  - the **rank** of the documents in each results list;
  - the **probability** of a document being relevant, based on past performance.

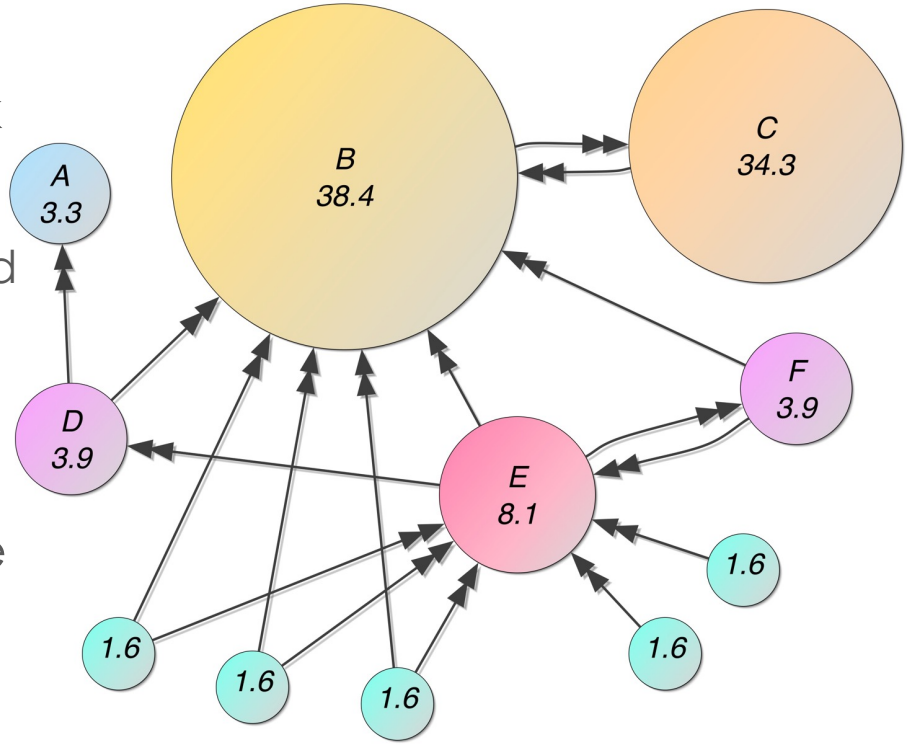


Learning to Rank

- After the initial search has been done by a classic ranked retrieval model (e.g. BM25, or a fused result from several rankers).
- Machine Learning used to re-rank the documents that have been retried.
  - **Aim**: improve the result by putting the most-relevant documents at the top of the ranked list of results.
- Takes into account much more information than just the initial ranking score.
  - Each piece of information used is known as a **feature** in Machine Learning.

PageRank: Document Importance

- One feature commonly used in Learning to Rank models is **PageRank**[1].
  - Developed by the founders of Google, and was a key part of Google's success.
- Measures the **importance** of a document based on the **link structure of the web**.
- A document is considered to be **important** if:
  - Many pages link to it; or
  - Other important pages link to it.



[1] Page, L., Brin, S., Motwani, R., and Winograd, T . *The PageRank Citation Ranking: Bringing Order to the Web*. Technical Report. Stanford InfoLab. 2009.

Learning to Rank: Features

- Some examples of features that can be used by Learning to Rank models[1]

- **Textual Features:**
  - Term occurrence/non-occurrence
  - Term frequency
  - Inverse Document Frequency
  - Document Length
  - Term Proximity

- **Non-Textual Features:**
  - PageRank
  - URL Depth
  - Document Quality
  - Readability
  - Sentiment
  - Query Clarity

[1] Metzler, D. ,and Croft, W. B., Linear Feature-Based Models for Information Retrieval, 2007. https://doi.org/10.1007/s10791-006-9019-z

Neural Language Models

- Bidirectional Encoder Representations from Transformers (BERT) is a deep-learning based Large Language Model (LLM) for Natural Language Processing.
  - Developed by Google and released in 2018[1].
  - In simple terms, it learns the relationships between words to aid document understanding
  - It has revolutionised NLP, and has had great success on many NLP tasks.
  - As at 2020, it is used for almost all English-language queries in Google's search engine[2].

[1] https://github.com/google-research/bert
[2] https://searchengineland.com/google-bert-used-on-almost-every-english-query-342193

What about ChatGPT?

- Probably the most famous LLM is now ChatGPT, released in 2022 based on GPT-3.5 by OpenAI.
  - Now ChatGPT-Plus is based on GPT-4, released in 2023.
  - Microsoft have integrated ChatGPT into Bing search engine (announced February 2023).
    - It will be interesting to see how users respond in the longer term, compared to traditional web search.
- LLMs are made by learning patterns and relationships between words/phrases from **enormous** quantities of text.
  - It is estimated to cost 27.5m RMB (4m USD) to train GPT-3, for example[1].
- How does ChatGPT it work?[2]

[1] https://www.cnbc.com/2023/03/13/chatgpt-and-generative-ai-are-booming-but-at-a-very-expensive-price.html
[2] https://www.zdnet.com/article/how-does-chatgpt-work/

What about ChatGPT?

- In the context of IR, LLMs such as ChatGPT (and others: e.g. Bard by Google and LLaMa by Meta) have several serious problems.
  - "In simple terms, these models figure out what word is likely to come next, given a set of words or a phrase. In doing so, they are able to generate sentences, paragraphs and even pages that correspond to a query from a user." [1]
  - It is **not designed to provide knowledge** and so it is often **confidently wrong**. The output can be plausible and believable, but might still be incorrect.
  - It cannot **cite its sources**, so you cannot find where it got its information from. If you ask it to provide citations, it will provide something that seems to be correct, but these will be artificial, because it is trained to know **what a citation looks like**, and not to understand any knowledge about a citation.

[1] https://theconversation.com/ai-information-retrieval-a-search-engine-researcher-explains-the-promise-and-peril-of-letting-chatgpt-and-its-cousins-search-the-web-for-you-200875

Conclusion

- The Classical IR Pipeline has several limitations with regard to modern IR.
- Several techniques are available that can be used to help improve the quality of retrieval, but **response time is key** given the huge amounts of data involved.
- In the next few lectures, we will examine a few of these in more detail.