Extended Probabilistic Models

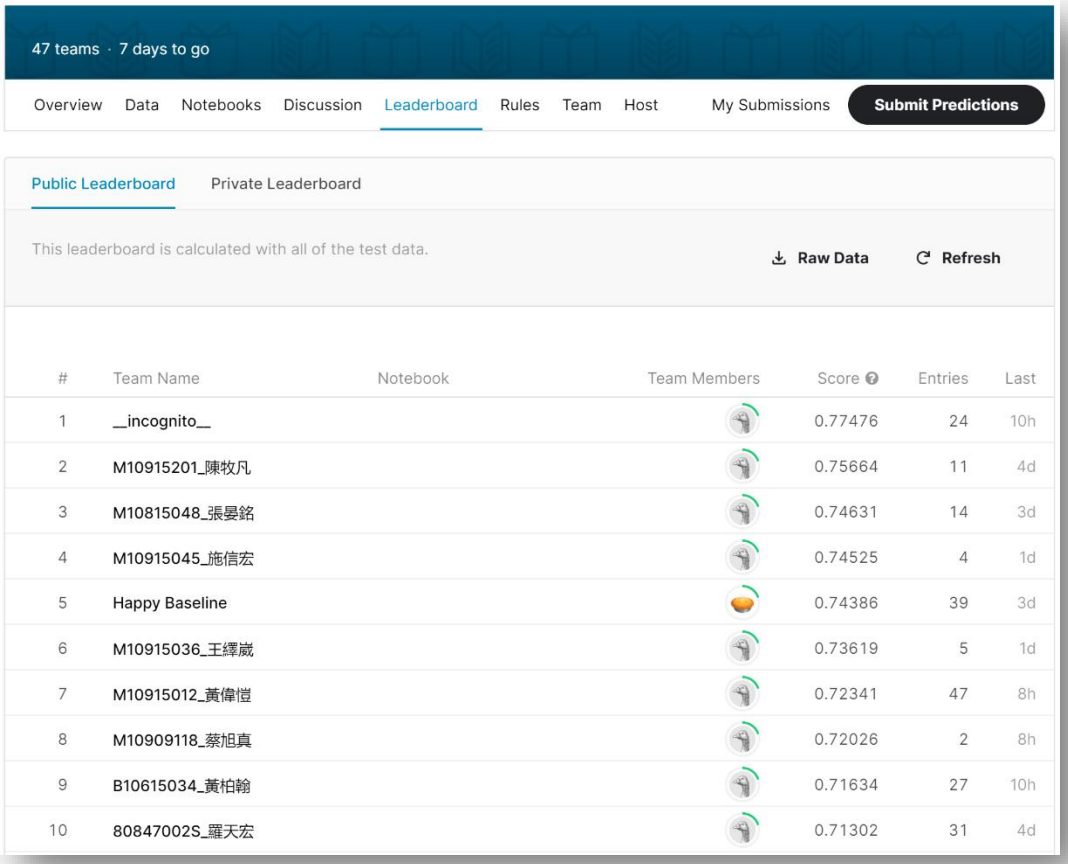
Kuan-Yu Chen (陳冠宇)

2020/10/23 @ TR-313, NTUST

Homework 1 – VSM

- In this project, we have
  - 50 Queries
  - 4191 Documents
  - Our goal is to implement a vector space model

$$sim(q, d_j) = cos(\theta) = \frac{\vec{q} \cdot \vec{d}_j}{|\vec{q}||\vec{d}_j|} = \frac{\sum_{w_i \in V} w_{i,q} \times w_{i,j}}{\sqrt{\sum_{w_i \in V} w_{i,q}^2} \times \sqrt{\sum_{w_i \in V} w_{i,j}^2}}$$



Review

- Boolean Model

- Probabilistic Model
  - Binary Independence Model

$$sim(d_j, q) \propto \sum_{w_i \in d_j \& w_i \in q} log \frac{P(w_i|R_q)}{1 - P(w_i|R_q)} + log \frac{1 - P(w_i|\bar{R}_q)}{P(w_i|\bar{R}_q)}$$

  - Robertson-Sparck Jones Equation

$$sim(d_j, q) \equiv \sum_{w_i \in d_j \& w_i \in q} log \left(\frac{r_i + 0.5}{R_q - r_i + 0.5} \cdot \frac{N - R_q - n_i + r_i + 0.5}{n_i - r_i + 0.5} \right)$$

$$\equiv \sum_{w_i \in d_j \& w_i \in q} log \left(\frac{N - n_i + 0.5}{n_i + 0.5} \right)$$

- TF-IDF
  - Term Frequency
  - Inverse Document Frequency
    - Overlap Score Model $sim(q, d_j) = \sum_{w_i \in q} TF - IDF_{i,j}$
    - Vector Space Model $sim(q, d_j) = cos(\theta) = \frac{\vec{q} \cdot \vec{d}_j}{|\vec{q}||\vec{d}_j|} = \frac{\sum_{w_i \in V} k_{i,q} \times k_{i,j}}{\sqrt{\sum_{w_i \in V} k_{i,q}^2} \times \sqrt{\sum_{w_i \in V} k_{i,j}^2}}$

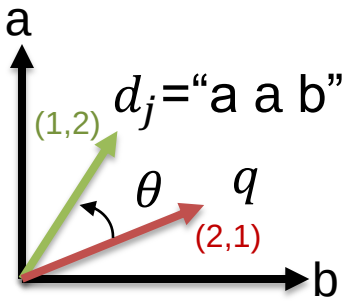
Document Length Normalization

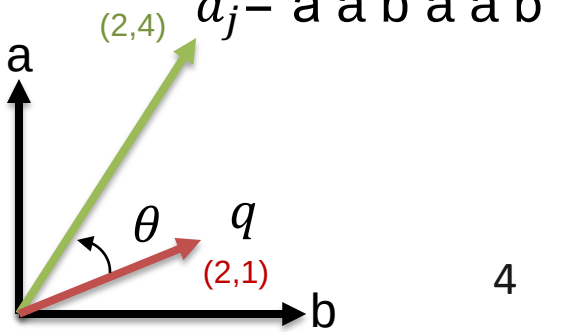
- Longer documents can broadly be lumped into two categories
  - Verbose documents that essentially **repeat the same content**
    - The longer the document, the more the information?
  - Documents **covering multiple different topics**
    - The term frequency cannot really reveal the document

$$sim_{Probabilistic}(d_j, q) \equiv \sum_{w_i \in d_j \& w_i \in q} log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

$$sim_{VSM}(q, d_j) \equiv cos(\theta) = \frac{\vec{q} \cdot \vec{d}_j}{|\vec{q}||\vec{d}_j|} = \frac{\sum_{w_i \in V} w_{i,q} \times w_{i,j}}{\sqrt{\sum_{w_i \in V} w_{i,q}^2} \times \sqrt{\sum_{w_i \in V} w_{i,j}^2}}$$

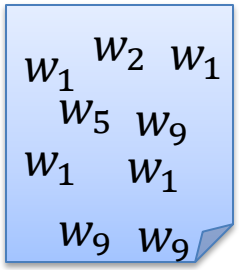
Scheme	Document Term Weight	Query Term Weight
1	$tf_{i,j} \times log\frac{N}{n_i}$	$\left(0.5 + 0.5\frac{tf_{i,q}}{max_i(tf_{i,q})}\right) \times log\frac{N}{n_i}$
2	$1 + tf_{i,j}$	$log\left(1 + \frac{N}{n_i}\right)$
3	$(1 + tf_{i,j}) \times log\frac{N}{n_i}$	$(1 + tf_{i,q}) \times log\frac{N}{n_i}$



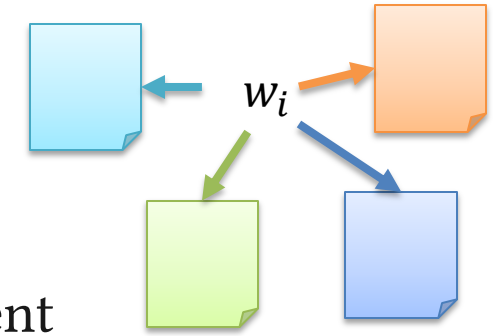


TF, IDF, and Document Length

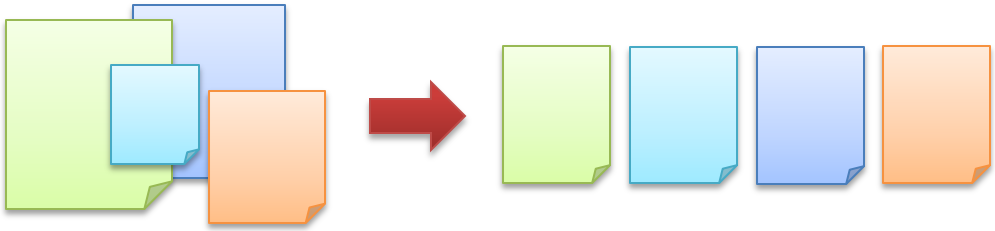
- Term Frequency
  - Based on the observations, high frequency terms are important for describing documents



- Inverse Document Frequency
  - IDF is used to demonstrate the specification of the term



- Document Length Normalization
  - Unify the information quantity of each document



Best Match Models

Best Match Models

- Best Match models were created as the results of a series of experiments on variations of the probabilistic model

- A good term weighting is based on three principles
  - inverse document frequency
  - term frequency
  - document length normalization

- The classic probabilistic model covers only the first of these principles

$$sim_{Probabilistic}(d_j, q) \propto \sum_{w_i \in d_j \& w_i \in q} log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

  - This reasoning led to a series of experiments with the Okapi system, which led to the "BM25" ranking formula

Best Match 1 – BM1

- At first, the Okapi system used the Equation below as ranking formula

$$sim_{BM1}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

- The equation is used in the classic probabilistic model
  - No relevance information can be leverage to estimate a fully probabilistic estimation
  - Consequently, a simple variant is derived

- It was referred to as the BM1 formula

$$sim(d_j, q) \propto \sum_{w_i \in d_j \& w_i \in q} log\left(\frac{r_i + 0.5}{R_q - r_i + 0.5} \cdot \frac{N - R_q - n_i + r_i + 0.5}{n_i - r_i + 0.5}\right)$$
$$\approx \sum_{w_i \in d_j \& w_i \in q} log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

Best Match 15 – 1

- The first idea for improving the ranking was to introduce a **term-frequency** factor $\mathcal{F}_{i,j}$ in the BM1 formula
  - For document d_j

$$\mathcal{F}_{i,j} = S_1 \times \frac{tf_{i,j}}{K_1 + tf_{i,j}}$$

  - For query q

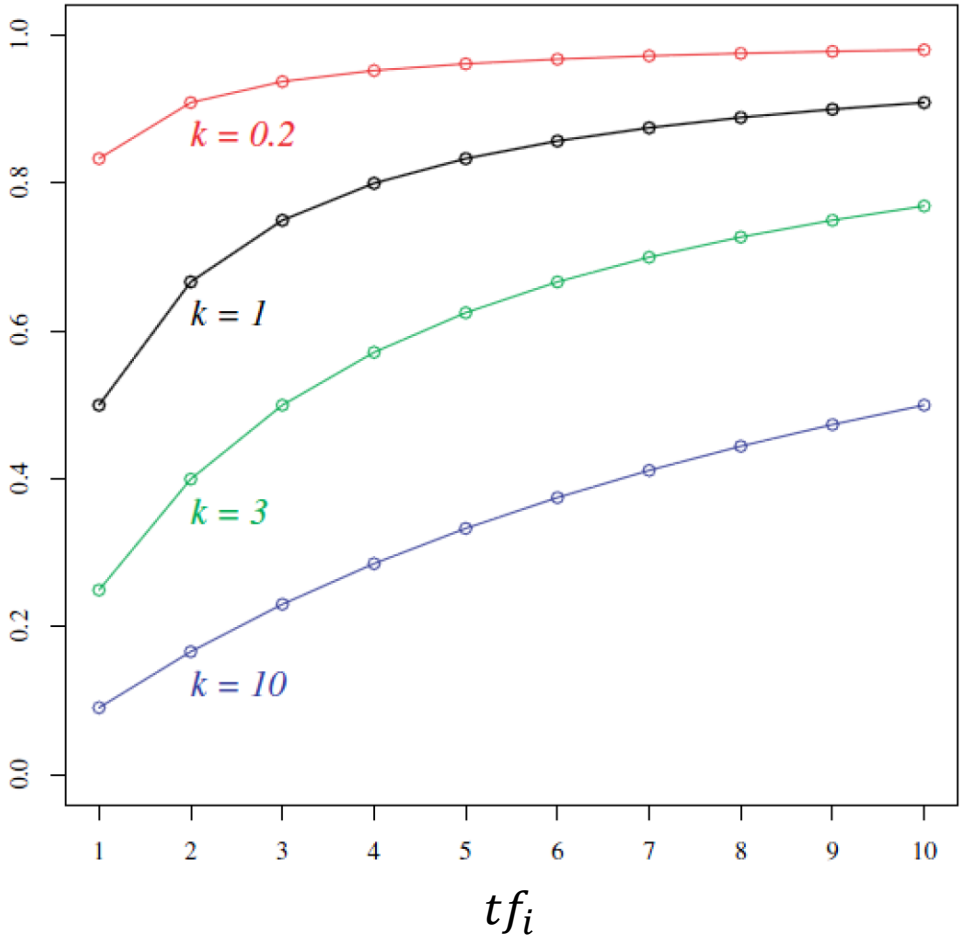
$$\mathcal{F}_{i,q} = S_3 \times \frac{tf_{i,q}}{K_3 + tf_{i,q}}$$

    - $tf_{i,j}$ (and $tf_{i,q}$) is the frequency of term w_i within d_j (and q)
    - K_1 and K_3 are constants setup experimentally for each collection
    - S_1 and S_3 are scaling constants, normally set to $S_1 = K_1 + 1$ and $S_3 = K_3 + 1$

Best Match 15 – 2

- $\frac{tf_i}{K+tf_i}$ is a saturation function
  - The resulting score is between 0 and 1

$\frac{tf_i}{K + tf_i}$



Best Match 15 – 3

- Next, a correction factor $G_{j,q}$ dependent on the document and query lengths was introduced

$$G_{j,q} = K_2 \times len(q) \times \frac{avg_{doclen} - len(d_j)}{avg_{doclen} + len(d_j)}$$

  - $len(q)$ is the query length (number of terms in the query)
  - $len(d_j)$ is the document length
  - avg_{doclen} is the average length of documents in the collection
  - K_2 is a constant

Best Match 15 – 4

- To put everything together, we can obtain the BM15 model

$$sim_{BM15}(d_j, q) \equiv G_{j,q} + \sum_{w_i \in \{d_j \cap q\}} \mathcal{F}_{i,j} \times \mathcal{F}_{i,q} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

$$\boldsymbol{sim_{BM15}(d_j, q) \equiv K_2 \times \frac{len(q) \times (avg_{doclen} - len(d_j))}{avg_{doclen} + len(d_j)} + \sum_{w_i \in \{d_j \cap q\}} \frac{S_1 \times tf_{i,j}}{K_1 + tf_{i,j}} \times \frac{S_3 \times tf_{i,q}}{K_3 + tf_{i,q}} \times \log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)}$$

Tunable Parameters
Document Length Normalization (Correction Factor)
Term Frequency
Inverse Document Frequency

Best Match 11

- A variant was to modify the $\mathcal{F}_{i,j}$ factor by adding **document length normalization** to it

$$\mathcal{F}'_{i,j} = S_1 \times \frac{tf_{i,j}}{\frac{K_1 \times len(d_j)}{avg_{doclen}} + tf_{i,j}}$$

  - The model is named BM11

$$sim_{BM11}(d_j, q) \equiv G_{j,q} + \sum_{w_i \in \{d_j \cap q\}} \mathcal{F}'_{i,j} \times \mathcal{F}_{i,q} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

$$\boldsymbol{sim_{BM11}(d_j, q)}$$

$$\equiv \boldsymbol{K_2} \times \frac{\boldsymbol{len(q) \times (avg_{doclen} - len(d_j))}}{\boldsymbol{avg_{doclen} + len(d_j)}} + \sum_{\boldsymbol{w_i \in \{d_j \cap q\}}} \frac{\boldsymbol{S_1 \times tf_{i,j}}}{\boldsymbol{K_1 \times \frac{len(d_j)}{avg_{doclen}} + tf_{i,j}}} \times \frac{\boldsymbol{S_3 \times tf_{i,q}}}{\boldsymbol{K_3 + tf_{i,q}}} \times \boldsymbol{\log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)}$$

Tunable Parameters

Document Length Normalization (Correction Factor)

Term Frequency

Inverse Document Frequency

BM1, BM15, and BM11

- Introduction of these three factors led to various BM (Best Matching) formulas, as follows:

$$sim_{BM1}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

$$sim_{BM15}(d_j, q) \equiv G_{j,q} + \sum_{w_i \in \{d_j \cap q\}} \mathcal{F}_{i,j} \times \mathcal{F}_{i,q} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

$$sim_{BM11}(d_j, q) \equiv G_{j,q} + \sum_{w_i \in \{d_j \cap q\}} \mathcal{F}'_{i,j} \times \mathcal{F}_{i,q} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

- Experiments using TREC data have shown that BM11 outperforms BM15

Empirical Considerations

- Several empirical considerations have been discussed
  - Empirical evidence suggests that a best value of K_2 is 0, which eliminates the $G_{j,q}$ factor from these equations

$$sim_{BM11}(d_j, q) \equiv K_2 \times \frac{len(q) \times (avg_{doclen} - len(d_j))}{avg_{doclen} + len(d_j)} + \sum_{w_i \in \{d_j \cap q\}} \frac{S_1 \times tf_{i,j}}{K_1 \times \frac{len(d_j)}{avg_{doclen}} + tf_{i,j}} \times \frac{S_3 \times tf_{i,q}}{K_3 + tf_{i,q}} \times \log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

  - Further, good estimates for the scaling constants S_1 and S_3 are $K_1 + 1$ and $K_3 + 1$, respectively

  - Empirical evidence also suggests that making K_3 very large is better
    - When K_3 is very large, $\mathcal{F}_{i,q}$ factor is reduced simply to $tf_{i,q}$

$$\mathcal{F}_{i,q} = S_3 \times \frac{tf_{i,q}}{K_3 + tf_{i,q}} = \frac{(K_3 + 1)}{K_3 + tf_{i,q}} \times tf_{i,q} \approx tf_{i,q}$$

BM1, BM15, and BM11 Formulas

- By following the considerations, BM models lead to simpler equations

$$sim_{BM1}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

$$sim_{BM15}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} \frac{(K_1 + 1) \times tf_{i,j}}{K_1 + tf_{i,j}} \times \frac{(K_3 + 1) \times tf_{i,q}}{K_3 + tf_{i,q}} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

$$sim_{BM11}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} \frac{(K_1 + 1) \times tf_{i,j}}{\frac{K_1 \times len(d_j)}{avg_{doclen}} + tf_{i,j}} \times \frac{(K_3 + 1) \times tf_{i,q}}{K_3 + tf_{i,q}} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

Best Match 25 – BM25

- The only difference between BM15 and BM 11 is the estimation of the term frequency

$$\mathcal{F}_{i,j}^{BM15} = \frac{(K_1 + 1) \times tf_{i,j}}{K_1 + tf_{i,j}} \qquad \mathcal{F}_{i,j}^{BM11} = \frac{(K_1 + 1) \times tf_{i,j}}{\frac{K_1 \times len(d_j)}{avg_{doclen}} + tf_{i,j}}$$

- BM25 is proposed to combine BM15 and BM11

$$\mathcal{F}_{i,j}^{BM25} = \frac{(K_1 + 1) \times tf_{i,j}}{K_1 \left[(1 - b) + b \times \frac{len(d_j)}{avg_{doclen}}\right] + tf_{i,j}}$$

  - b is a constant with values in the interval [0,1]
    - If $b = 0$, it reduces to the BM15 term frequency factor
    - If $b = 1$, it reduces to the BM11 term frequency factor
    - For values of b between 0 and 1, the equation provides a combination of BM11 with BM15

BM25 Formula

- To sum up, the BM25 model can be written as:

$$sim_{BM25}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} \frac{(K_1 + 1) \times tf_{i,j}}{K_1 \left[(1 - b) + b \times \frac{len(d_j)}{avg_{doclen}} \right] + tf_{i,j}} \times \frac{(K_3 + 1) \times tf_{i,q}}{K_3 + tf_{i,q}} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5} \right)$$

Tunable Parameters
Document Length Normalization
Term Frequency
Inverse Document Frequency

  - b should be kept closer to 1 to emphasize the document length normalization effect present in the BM11 formula
    - $b = 0.75$ is a reasonable assumption
  - Constants values (i.e., K_1, K_3, and b) can be fine tuned for particular collections through proper experimentation

Further Analysis – 1

- The Okapi BM25 is a state-of-the-art retrieval function for nearly two decades

- The formula can be presented:
  - BM25 weighting

$$\sum_{w_i \in \{d_j \cap q\}} \frac{(K_1+1) \times tf_{i,j}}{K_1 \left[(1-b) + b \times \frac{len(d_j)}{avg_{doclen}} \right] + tf_{i,j}} \times \frac{(K_3+1) \times tf_{i,q}}{K_3 + tf_{i,q}} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5} \right)$$

Document term weighting

Query term weighting

Discriminative power

Further Analysis – 2

- The key component of BM25 contributing to its success is its term frequency normalization formula:

$$\frac{(K_1+1)\times tf_{i,j}}{K_1\left[(1-b)+b\times\frac{len(d_j)}{avg_{doclen}}\right]+tf_{i,j}}=\frac{(K_1+1)\times tf'_{i,j}}{K_1+tf'_{i,j}}$$

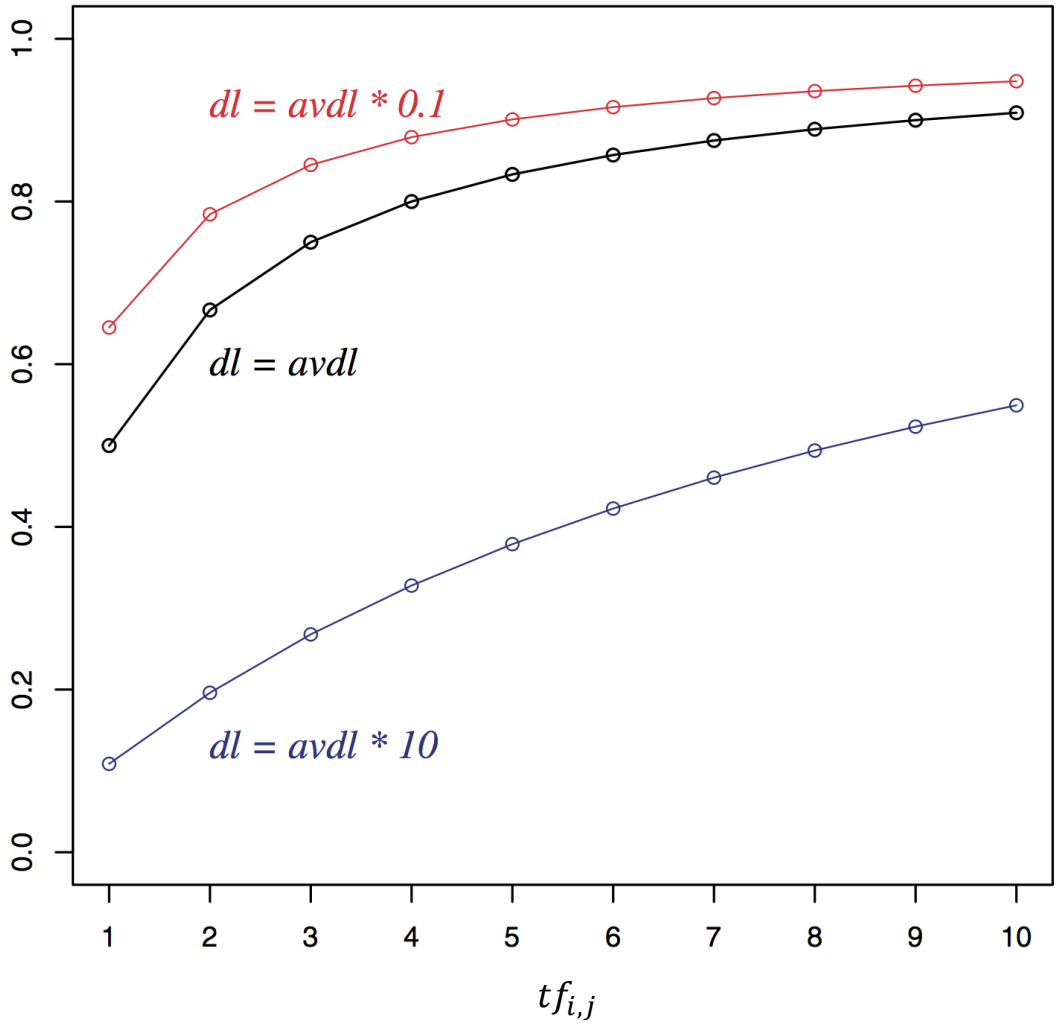
  - $tf'_{i,j}$ is the normalized term frequency by document length using **pivoted length normalization**

$$tf'_{i,j}=\frac{tf_{i,j}}{1-b+b\times\frac{len(d_j)}{avg_{doclen}}}$$

$$sim_{BM25}(d_j,q)\equiv\sum_{w_i\in\{d_j\cap q\}}\frac{(K_1+1)\times tf'_{i,j}}{K_1+tf'_{i,j}}\times\frac{(K_3+1)\times tf_{i,q}}{K_3+tf_{i,q}}\times log\left(\frac{N-n_i+0.5}{n_i+0.5}\right)$$

Further Analysis – 3

$$\frac{tf_{i,j}}{K_1\left[(1-b)+b\times\frac{len(d_j)}{avg_{doclen}}\right]+tf_{i,j}}$$



Pros and Cons

- Advantages
  - Unlike the probabilistic model, the BM25 formula can be computed without relevance information
  - There is consensus that BM25 outperforms the classic vector model for general collections

$$sim_{BM25}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} \frac{(K_1 + 1) \times tf'_{i,j}}{K_1 + tf'_{i,j}} \times \frac{(K_3 + 1) \times tf_{i,q}}{K_3 + tf_{i,q}} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5}\right)$$

- Disadvantages
  - When a document is very long, we can see that $tf'_{i,j}$ could be very small and approach zero!
    - The presence of w_i in a very long document fails to differentiate clearly from other documents where w_i is absent
    - This suggests that **those very long documents can be overly penalized**

$$tf'_{i,j} = \frac{tf_{i,j}}{1 - b + b \times \frac{len(d_j)}{avg_{doclen}}}$$

Y. Lv and C.X. Zhai. 2011. When documents are very long, BM25 fails!. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR '11). ACM, New York, NY, USA, 1103-1104.

Boosting Very Long Documents

- In order to avoid overly-penalizing very long documents, one heuristic way to achieve this goal is to define:

$$\begin{cases} \dfrac{(K_1+1)\times\left[tf'_{i,j}+\delta\right]}{K_1+\left[tf'_{i,j}+\delta\right]} & ,if\ \ tf'_{i,j}>0 \\ 0 & ,otherwise \end{cases}$$

$$sim_{BM25L}(d_j,q) \equiv \sum_{w_i\in\{d_j\cap q\}} \frac{(K_1+1)\times\left[tf'_{i,j}+\delta\right]}{K_1+\left[tf'_{i,j}+\delta\right]} \times \frac{(K_3+1)\times tf_{i,q}}{K_3+tf_{i,q}} \times log\left(\frac{N-n_i+0.5}{n_i+0.5}\right)$$

  - A shifted version by adding a shift parameter $\delta > 0$
  - We can notice that the modification has a **positive lower bound** for $tf'_{i,j} > 0$

$$\frac{(K_1+1)\times\delta}{K_1+\delta}$$

$$tf'_{i,j} = \frac{tf_{i,j}}{1-b+b\times\dfrac{len(d_j)}{avg_{doclen}}}$$

Language Models

Language Modeling

- A goal of statistical language modeling is to learn the joint probability function of sequences of words in a language

$$P(w_1, w_2, \dots, w_T)$$

- A statistical model of language can be represented by the conditional probability of the next word given all the previous ones (**chain rule**)

$$P(w_1, w_2, \dots, w_T) = \prod_{t=1}^{T} P(w_t | w_1, w_2, \dots, w_{t-1})$$

$$\approx \prod_{t=1}^{T} P(w_t | w_{t-n+1}, \dots, w_{t-1})$$

  - Such statistical language models have already been found useful in many technological applications involving natural language

N-gram

- Assume words (terms)
  - are independent of each other
  - follow a multinomial distribution

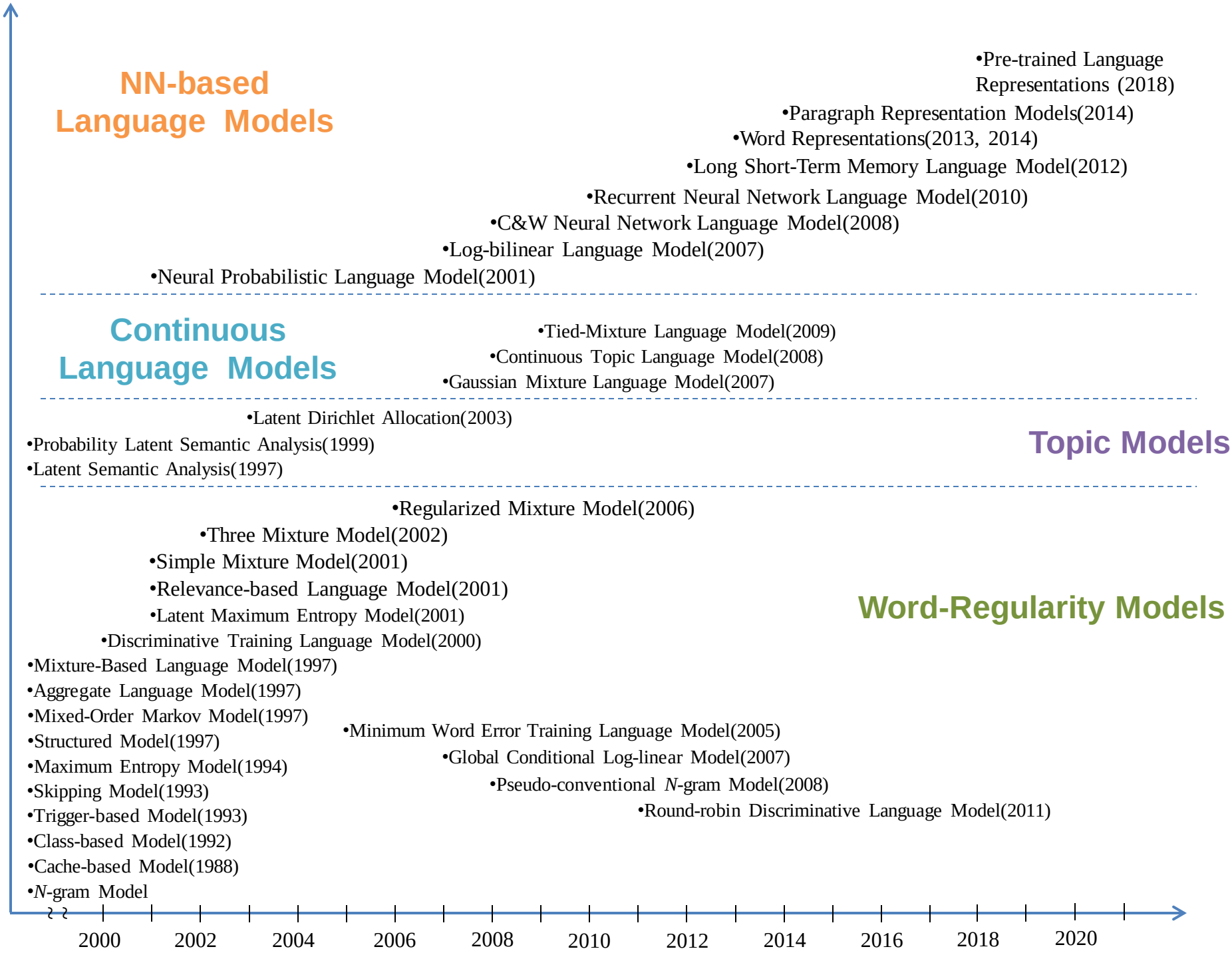
- Unigram
  - Each word occurs independently of the other words
  - The so-called "bag-of-words" model

$$P(w_1, w_2, \dots, w_T) = P(w_1) \cdot P(w_2) \cdots P(w_T) = \prod_{t=1}^{T} P(w_t)$$

- Bigram

$$P(w_1, w_2, \dots, w_T) = P(w_1)P(w_2|w_1) \cdots P(w_T|w_{T-1}) = P(w_1) \prod_{t=2}^{T} P(w_t|w_{t-1})$$

- Unigram model is the most popular choice in IR
  - IR does not directly depend on the structure of sentences


NN-based Language Models
•Pre-trained Language Representations (2018)
•Paragraph Representation Models(2014)
•Word Representations(2013, 2014)
•Long Short-Term Memory Language Model(2012)
•Recurrent Neural Network Language Model(2010)
•C&W Neural Network Language Model(2008)
•Log-bilinear Language Model(2007)
•Neural Probabilistic Language Model(2001)
Continuous Language Models
•Tied-Mixture Language Model(2009)
•Continuous Topic Language Model(2008)
•Gaussian Mixture Language Model(2007)
•Latent Dirichlet Allocation(2003)
•Probability Latent Semantic Analysis(1999)
•Latent Semantic Analysis(1997)
Topic Models
•Regularized Mixture Model(2006)
•Three Mixture Model(2002)
•Simple Mixture Model(2001)
•Relevance-based Language Model(2001)
•Latent Maximum Entropy Model(2001)
•Discriminative Training Language Model(2000)
•Mixture-Based Language Model(1997)
•Aggregate Language Model(1997)
•Mixed-Order Markov Model(1997)
•Structured Model(1997)
•Maximum Entropy Model(1994)
•Skipping Model(1993)
•Trigger-based Model(1993)
•Class-based Model(1992)
•Cache-based Model(1988)
•N-gram Model
Word-Regularity Models
•Minimum Word Error Training Language Model(2005)
•Global Conditional Log-linear Model(2007)
•Pseudo-conventional N-gram Model(2008)
•Round-robin Discriminative Language Model(2011)
2000
2002
2004
2006
2008
2010
2012
2014
2016
2018
2020

NN-based Language Models




Word/ Paragraph/ Language Representations (2013~)

Neural Network Language Models (2001~)

Continuous Language Models

Continuous Language Models (2007~2009)

Topic Models (1997~2003)


Topic Models

Query Language Models (2001~2006)


Word-Regularity Models

Discriminative Language Models (2000~2011)




Word-Regularity Models (~1997)

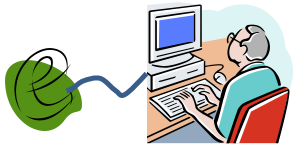
2000 2002 2004 2006 2008 2010 2012 2014 2016 2018 2020

https://images.app.goo.gl/8D6Hoe6mxhdVvDPh9

SESAME STREET

- The Sesame Street almost dominates the NLP community now!

Language Modeling for Retrieval



- (Statistical) language models (LM) have been widely used for **speech recognition** and **language (machine) translation** for more than thirty years

- However, their use for information retrieval started only in 1998
  - Basically, a query is considered generated from an "ideal" document that satisfies the information need
    - Query-Likelihood Measure
    - KL-Divergence Measure

Query Likelihood Measure

- In the query likelihood retrieval model, we rank documents by the probability that the query could be generated by the document language model

$$P(d_j|q) = \frac{P(q|d_j)P(d_j)}{P(q)} \propto P(q|d_j)P(d_j)$$

$$\approx P(q|d_j) \approx \prod_{i=1}^{|q|} P(w_i|d_j)$$

Document Model

  - The user has a prototype (ideal) document in mind, and generates a query based on words that appear in this document
  - A document is treated as a model to predict (generate) the query

Document Model – Unigram.

- Use each document itself a sample for estimating its corresponding unigram model
  - The unigram model is a multinomial distribution
  - If Maximum Likelihood Estimation (MLE) is adopted

$$P(w_i|d_j) = \frac{c(w_i, d_j)}{|d_j|}$$

d_j

w_2	w_1	w_3
w_1	w_1	w_2
w_3	w_3	w_3
w_1	w_2	w_3

$$P(w_1|d_j) = \frac{4}{12}$$

$$P(w_2|d_j) = \frac{3}{12}$$

$$P(w_3|d_j) = \frac{5}{12}$$

$$\begin{aligned} P(q|d_j) &= P(w_1, w_2, w_2|d_j) \\ &= P(w_1|d_j)P(w_2|d_j)P(w_2|d_j) \\ &= \frac{4}{12} \times \frac{3}{12} \times \frac{3}{12} \end{aligned}$$

Document Model – Unigram..

- Use each document itself a sample for estimating its corresponding unigram model
  - The unigram model is a multinomial distribution
  - If Maximum Likelihood Estimation (MLE) is adopted

$$P(w_i|d_j) = \frac{c(w_i, d_j)}{|d_j|}$$

d_j

w_2	w_1	w_3
w_1	w_1	w_2
w_3	w_3	w_3
w_1	w_2	w_3

$$P(w_1|d_j) = \frac{4}{12}$$

$$P(w_2|d_j) = \frac{3}{12}$$

$$P(w_3|d_j) = \frac{5}{12}$$

$$P(w_4|d_j) = \frac{0}{12}$$

$$P(q|d_j) = P(w_1, w_1, w_4|d_j)$$
$$= P(w_1|d_j)P(w_1|d_j)P(w_4|d_j)$$
$$= \frac{4}{12} \times \frac{4}{12} \times 0 = 0$$

Zero-probability Problem!
Data Sparseness

Smoothing by Background Model.

- Smooth the document-specific unigram model with a collection model
  - Usually named the "background model"
  - The background model can be estimated in a similar way as what we do for the document unigram model

$$P(w_i|BG) = \frac{c(w_i, collection)}{|collection|}$$

- Two representative language model smoothing methods
  - Linear Interpolation (Jelinek-Mercer smoothing)

$$P'(w_i|d_j) = \lambda \cdot P(w_i|d_j) + (1-\lambda) \cdot P(w_i|BG)$$

  - Bayesian Smoothing with Dirichlet Prior

$$P'(w_i|d_j) = \frac{c(w_i, d_j) + \mu \cdot P(w_i|BG)}{|d_j| + \mu}$$

Smoothing by Background Model..

- Linear Interpolation (Jelinek-Mercer smoothing) is the popular one

$$P'(w_i|d_j) = \lambda \cdot P(w_i|d_j) + (1-\lambda) \cdot P(w_i|BG)$$

- The role of the background model
  - Help to solve zero-probability problem
  - Play a role as IDF?
  - Not clear!? But it is essential to the good performance

Smoothing and IDF

- Smoothing acts as the IDF factor

$$\log P'(q|d_j) \approx \log \prod_{i=1}^{|q|} P'(w_i|d_j) = \sum_{w_i \in V} \log P'(w_i|d_j)^{c(w_i,q)}$$

$$= \sum_{w_i \in d_j} \log P'(w_i|d_j)^{c(w_i,q)} + \sum_{w_i \notin d_j} \log P'(w_i|d_j)^{c(w_i,q)}$$

$$= \sum_{w_i \in d_j} \log\left[\lambda \cdot P(w_i|d_j) + (1-\lambda) \cdot P(w_i|BG)\right]^{c(w_i,q)} + \sum_{w_i \notin d_j} \log[(1-\lambda) \cdot P(w_i|BG)]^{c(w_i,q)}$$

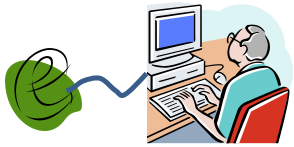
$$= \sum_{w_i \in d_j} \log\left[\lambda \cdot P(w_i|d_j) + (1-\lambda) \cdot P(w_i|BG)\right]^{c(w_i,q)}$$

$$+ \sum_{w_i \in V} \log[(1-\lambda) \cdot P(w_i|BG)]^{c(w_i,q)} - \sum_{w_i \in d_j} \log[(1-\lambda) \cdot P(w_i|BG)]^{c(w_i,q)}$$

$$= \sum_{w_i \in d_j} \log\left(\frac{\lambda \cdot P(w_i|d_j) + (1-\lambda) \cdot P(w_i|BG)}{(1-\lambda) \cdot P(w_i|BG)}\right)^{c(w_i,q)}$$

$$+ \sum_{w_i \in V} \log[(1-\lambda) \cdot P(w_i|BG)]^{c(w_i,q)}$$

KL-Divergence Measure



- Another basic formulation of LM for IR is the Kullback-Leibler (KL)-Divergence measure

$$KL(q||d_j) = \sum_{w \in V} P(w|q) log \frac{P(w|q)}{P(w|d_j)} \propto - \sum_{w \in V} P(w|q) log P(w|d_j)$$

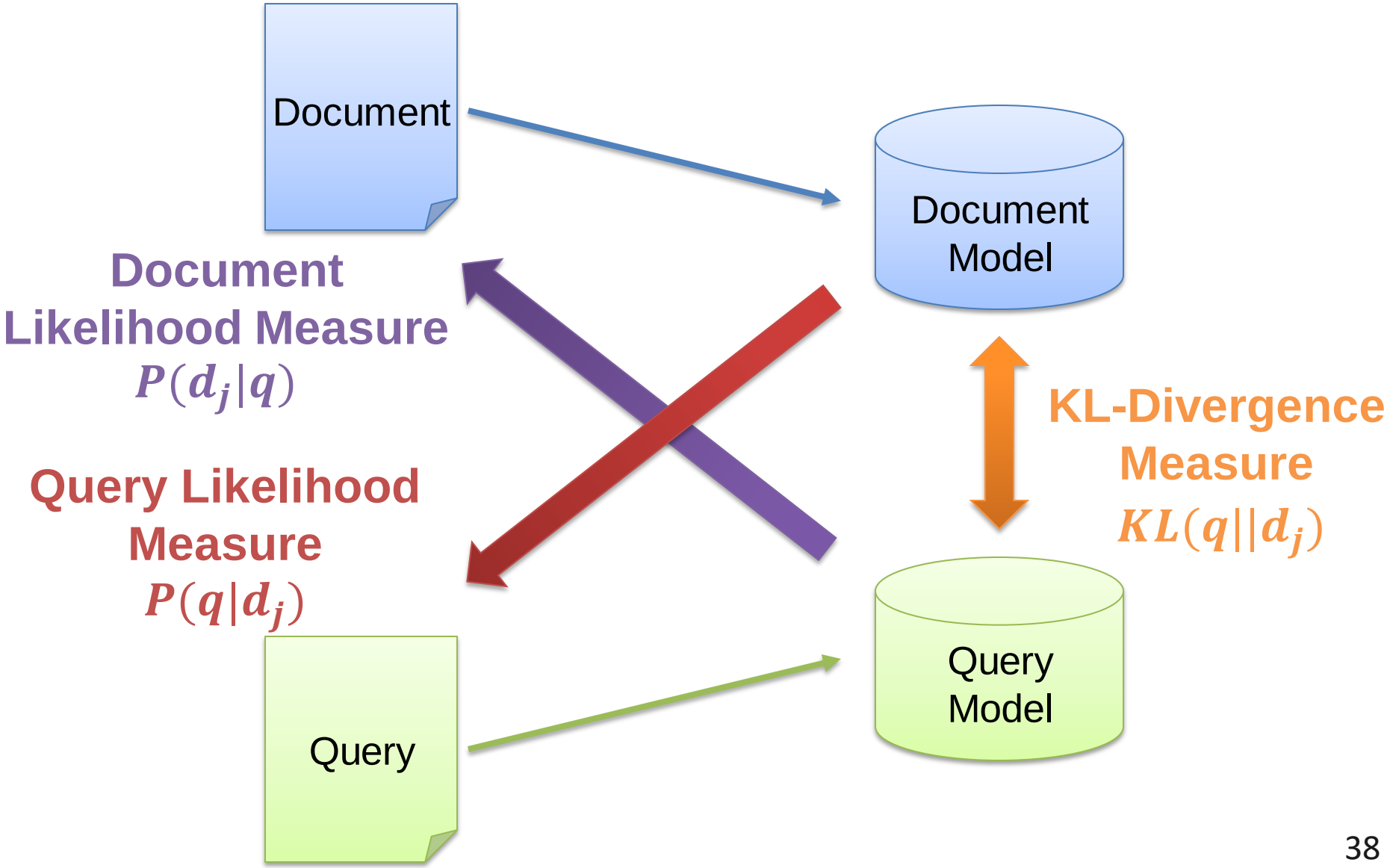
  - A query is treated as a **probabilistic model** rather than simply an **observation**
  - KL-divergence supports us to achieve a better result by considering **both** query and document models

- KLM can be degenerated to QLM

$$KL(q||d_j) \propto - \sum_{w \in V} P(w|q) log P(w|d_j) = - \sum_{w \in V} log P(w|d_j)^{\frac{c(w,q)}{|q|}}$$

$$\propto - \sum_{w \in V} log P(w|d_j)^{c(w,q)} = -log P(q|d_j)$$

Three Ways of LM Approaches for IR



The Evolution





1998 Language Model Approach

1994 Best Match Models (Okapi Systems)

Stephen Robertson, 1946-present

Gerard Salton, 1927-1995

1976 Probabilistic Model

Frederick Wilfrid Lancaster, 1933-2013

1975 Vector Space Model

Karen Spärck Jones, 1935-2007

1973 Boolean Model

Hans Peter Luhn, 1896-1964

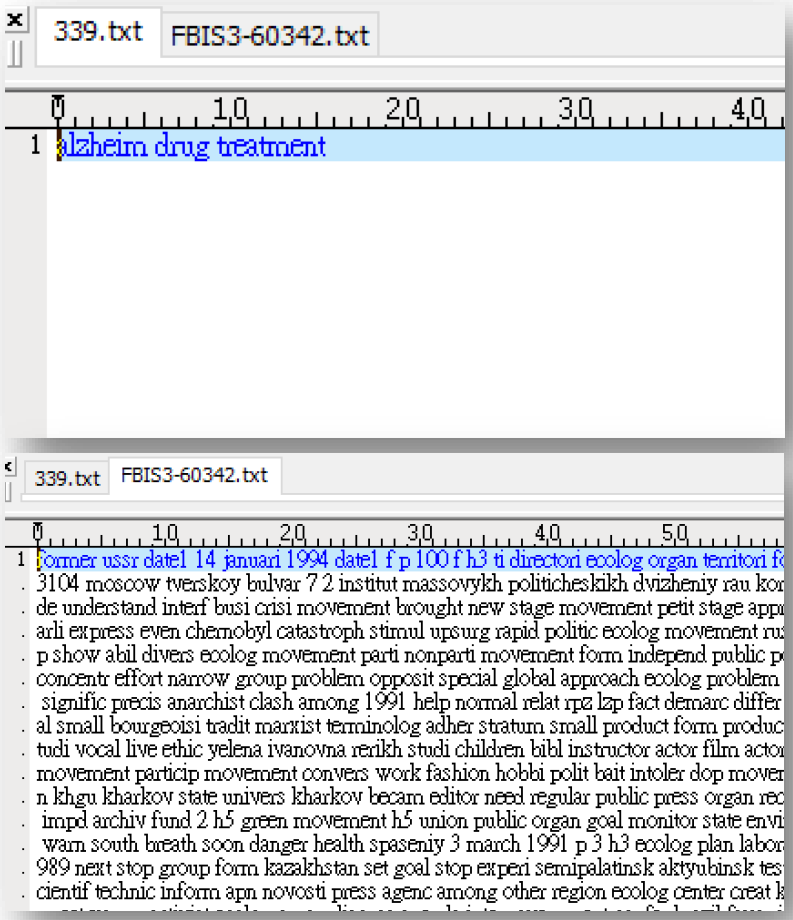
1972 Inverse Document Frequency

1957 Term Frequency

Homework 2 – Best Match Models

Homework 2 - Description.

- In this project, we have
  - 50 Queries
  - 4191 Documents
  - Our goal is to implement a BM model
    - BM1
    - BM15
    - BM11
    - BM25
    - BM25L

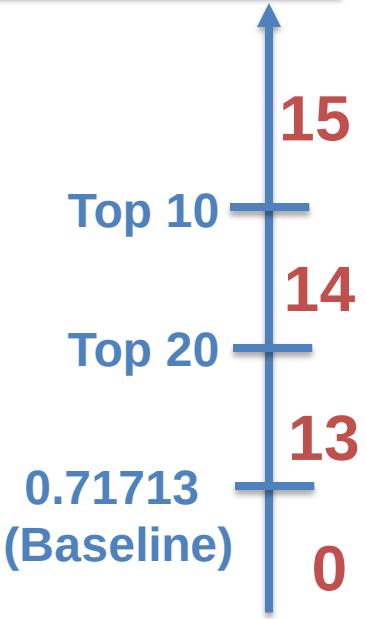


$$sim_{BM25}(d_j, q) \equiv \sum_{w_i \in \{d_j \cap q\}} \frac{(K_1 + 1) \times tf_{i,j}}{K_1 \left[(1 - b) + b \times \frac{len(d_j)}{avg_{doclen}} \right] + tf_{i,j}} \times \frac{(K_3 + 1) \times tf_{i,q}}{K_3 + tf_{i,q}} \times log\left(\frac{N - n_i + 0.5}{n_i + 0.5} \right)$$

Homework 2 – Description..

- The evaluation measure is MAP
  - The **hard** deadline is 11/5 23:59
  - You point is depended on your performance!
  - Please submit a **report** and your **source codes** to the Moodle system, otherwise you will get 0 point
    - The report will be judged by TA, and the score is either 1 or 2
- You should
  - upload your answer file to kaggle
    - https://www.kaggle.com/t/cd59d965ad1449159533515e1c4a239c
    - The maximum number of daily submissions is 20
    - **Your team name is ID_Name**

M123456_陳冠宇



Homework 2 – Submission Format

Questions?

kychen@mail.ntust.edu.tw