DFR

Divergence From Randomness

Quelques slides sont empruntés de Baeza-Yates & Ribeiro-Neto

Divergence from Randomness (DFR)

- Proposed by Amati and Rijsbergen
- The idea is to compute term weights by measuring the divergence between a term distribution produced by a random process (within the collection) and the actual term distribution (within the document)
- Thus, the name **divergence from randomness**
- The model is based on two fundamental assumptions, as follows

DFR: First assumption

- Not all words are equally important for describing the content of the documents
- Words that carry little information are assumed to be **randomly distributed** over the whole document collection C
  - Given a term t_i, its probability distribution over the whole collection is referred to as $P(t_i|C)$
  - The amount of information associated with this distribution is given by $-\log P(t_i|C)$

DFR: Second assumption

- A complementary term distribution can be obtained by considering just the subset of documents that contain term t_i
- This subset is referred to as the **elite set**
- The corresponding probability distribution, computed with regard to document d_j, is referred to as $P(t_i|d_j)$
  - Smaller the probability of observing a term t_i in a document d_j, more rare and important is the term considered to be
- Thus, the amount of information associated with the term in the elite set is defined as $1 - P(t_i|d_j)$

DFR: term weighting

- Given these assumptions, the weight $w_{i,j}$ of a term t_i in a document dj is defined as

$$w_{i,j} = [-log\ P(t_i|C)] * [1 - P(t_i|d_j)]$$

- Two term distributions are considered: in the collection and in the subset of docs in which it occurs
- The rank $R(d_j , q)$ of a document d_j with regard to a query q is then computed as

$$R(d_j, q) = \sum_{t_i \in q} tf_{i,q} \times w_{i,j}$$

where $tf_{i,q}$ is the frequency of term t_i in the query

Distribution of terms in the collection: (Random Distribution)

- To compute the distribution of terms in the collection, distinct probability models can be considered

- Binomial distribution
  - To illustrate, consider a collection with 1000 documents and a term t_i that occurs 10 times in the collection
  - Then, the probability of observing 4 occurrences of term t_i in a document is given by

$$P(t_j|C) = \binom{n}{k} p^k (1-p)^{(n-k)} = \binom{10}{4} \left(\frac{1}{1000}\right)^4 \left(1 - \frac{1}{1000}\right)^6$$

Distribution of terms in the collection: (Binomial distribution)

- In general, let p = 1/N be the probability of observing a term in a document, where N is the number of docs
- The probability of observing $f_{i,j}$ occurrences of term t_i in document d_j is described by a binomial distribution:

$$P(t_j|C) = \binom{TF_i}{tf_{i,j}} p^{tf_{i,j}} (1-p)^{TF_i - tf_{i,j}}$$

$$\mathrm{TF_i} = \sum_{\mathrm{d_j} \in C} \mathrm{tf_{i,j}}$$

TF_i is the total frequency of term t_i in the collection (N documents)

- The average occurrence of term t is : $\lambda_i = TF_i / N$

Distribution of terms in the collection: Binomial approximation→ Poisson

- As N>30 et $p < 0.05$
- Under these conditions, we can approximate the binomial distribution by a Poisson process, which yields

$$P(t_j|C) = \frac{e^{-\lambda_i} \lambda_i^{tf_{i,j}}}{tf_{i,j}!}$$

Binomial approximation→ Poisson

- The amount of information associated with term t_i in the collection can then be computed as

$$-\log P(t_j|C) = -\log\left(\frac{e^{-\lambda_i}\,\lambda_i^{f_{i,j}}}{f_{i,j}!}\right)$$

$$\approx -f_{i,j}\log\lambda_i + \lambda_i\log e + \log(f_{i,j}!)$$

$$\approx f_{i,j}\log\left(\frac{f_{i,j}}{\lambda_i}\right) + \left(\lambda_i + \frac{1}{12f_{i,j}+1} - f_{i,j}\right)\log e$$

$$+\frac{1}{2}\log(2\pi f_{i,j})$$

- $f_{i,j}!$ was approximated by the **Stirling's formula**

$$f_{i,j}! \approx \sqrt{2\pi}\, f_{i,j}^{(f_{i,j}+0.5)}\, e^{-f_{i,j}}\, e^{(12f_{i,j}+1)^{-1}}$$

Distribution of terms in the collection: Bose-Enstein distibution

- Randomness Model can be estimated as Bose-Einstein distribution and approximate it by a geometric distribution:

$$P(t_j|C) = p(1-p)^{tf}$$

where $p = 1/(1 + \lambda_i)$ (estimation of λ_i)

- The amount of information associated with term t_i in the collection can then be computed as

$$-\log P(t_j|C) \approx -\log\left(\frac{1}{1+\lambda_i}\right) - f_{i,j} \times \log\left(\frac{\lambda_i}{1+\lambda_i}\right)$$

Distribution over the Elite documents

- The amount of information associated with term distribution in elite docs can be computed by using

  Laplace's law of succession

$$1 - P(t_j | d_j) = \frac{1}{tf_{i,j} + 1}$$

- Another possibility is to adopt the ratio of two Bernoulli processes, which yields

$$1 - P(t_j | d_j) = \frac{TF_i + 1}{n_i \times (tf_{i,j} + 1)}$$

- n_i is the number of documents in which the term occurs

Normalization

- These formulations do not take into account the length of the document d_j. This can be done by normalizing the term frequency $tf_{i,j}$
- Distinct normalizations can be used, such as

$$tf'_{i,j} = tf_{i,j} \times \frac{avg\_dl}{dl(d_j)}$$

or

$$tf'_{i,j} = tf_{i,j} \times \log(1 + \frac{avg\_dl}{dl(d_j)})$$

where avg_dl is the average document length in the collection and $dl(d_j)$ is the length of document d_j

Normalization

- To compute $w_{i,j}$ weights using normalized term frequencies, just substitute the factor $tf_{i,j}$ by $tf'_{i,j}$
- we consider that a same normalization is applied for computing $P(t_i|C)$ and $P(t_i|d_j)$
- By combining different forms of computing $P(t_i|C)$ and $P(t_i|d_j)$ with different normalizations, various ranking formulas can be produced