CSA4020

Multimedia Systems:
Adaptive Hypermedia Systems

Lecture 6:
Probabilistic Models of
Information Retrieval
Disadvantages of Statistical Model

• Same doc in different collections can have different degrees of similarity to the same query

A bit like deciding on whether a Honda Civic is a good purchase on its own merits vs. other members of family having a Merc.!

• Prefer to have impartial estimation of relevance to any query, irrespective of collection

Provided by Boolean IR, but measure is too narrow (if all query terms in doc then relevant, otherwise not)

We also know (from experience) that docs which do not necessarily contain the query terms may be relevant
More disadvantages

- A term which has a high $df$ is given a low weight in statistical systems

Could simply turn this into a high probability of observing the term...

... to *independently* determine the probability of relevance to a query of a document which contains that term
Binary Independence Retrieval Model

Fundamentals:

• Given a user query there is a set of documents which contains exactly the relevant documents and no other:

  the “ideal” answer set

• Given the ideal answer set, a query can be constructed that retrieves exactly this set

 Assumes that relevant documents are “clustered”, and that terms used adequately discriminate against non-relevant documents
• We do not know what are, in general, the properties of the ideal answer set

All we know is that documents have terms which “capture” semantic meaning

• When user submits a query, “guess” what might be the ideal answer set

• Allow user to interact, to describe the probabilistic description of the ideal answer set (by marking docs as relevant/non-relevant)
Probabilistic Principle: Assumption

• Given a user query $q$ and a document $d_j$ in the collection:

Estimate the probability that the user will find $d_j$ relevant to $q$

• Rank documents in order of their probability of relevance to the query (Probability Ranking Principle)

• Model assumes that probability of relevance depends on $q$ and document representations only

• Assumes that there is an ideal answer set!

• Assumes that terms are distributed differently in relevant and non-relevant documents
• Whether or not a document $x$ is retrieved depends on:

$$\Pr(\text{rel} \mid x): \text{the probability that } x \text{ is relevant}$$

$$\Pr(\text{nonrel} \mid x): \ldots x \text{ isn’t relevant}$$

• Document Ranking Function: document $x$ will be retrieved if

$$a_2 \Pr(\text{rel} \mid x) \geq a_1 \Pr(\text{nonrel} \mid x)$$

where $a_2$ is the cost of not retrieving a relevant document, and $a_1$ is the cost of retrieving a non-relevant document

$$g(x) = \frac{\Pr(\text{rel} \mid x)}{\Pr(\text{nonrel} \mid x)} \cdot \frac{a_1}{a_2} > 0$$

• If we knew $\Pr(\text{rel} \mid x)$ (or $\Pr(\text{nonrel} \mid x)$), solution would be trivial, but...
• Use Bayes Theorem to rewrite $\Pr(\text{rel}|x)$:

$$\Pr(\text{rel} | x) = \frac{\Pr(x | \text{rel})P(\text{rel})}{\Pr(x)}$$

$\Pr(x)$: probability of observing $x$

$P(\text{rel})$: a priori probability of relevance (ie, probability of observing a set of relevant documents)

$\Pr(x | \text{rel})$: probability that $x$ is in the given set of relevant docs

• Can do the same for $\Pr(\text{nonrel}|x)$

$$\Pr(\text{nonrel} | x) = \frac{\Pr(x | \text{nonrel})P(\text{nonrel})}{\Pr(x)}$$
• The document ranking function can be rewritten as:

\[
\log g(x) = \log \frac{\Pr(x \mid \text{rel}) \Pr(\text{rel})}{\Pr(x)} \cdot \frac{\Pr(x) \cdot \Pr(\text{nonrel}) \Pr(\text{nonrel})}{\Pr(x \mid \text{nonrel}) \Pr(\text{nonrel})}
\]

and simplified as:

\[
\log g(x) = \log \frac{\Pr(x \mid \text{rel})}{\Pr(x \mid \text{nonrel})} + \frac{\Pr(\text{rel})}{\Pr(\text{nonrel})}
\]

• \(\Pr(x \mid \text{rel})\) and \(\Pr(x \mid \text{rel})\) are still unknown!

• We will replace them in terms of keywords in the document
• **We assume that terms occur independently in relevant and nonrelevant documents...**

\[
\log g(x) = \prod_{i=1}^{t} \log \frac{\Pr(x_i | \text{rel})}{\Pr(x_i | \text{nonrel})} + C
\]

• **Pr**(*x*<sub><i>i</i></sub> | rel): probability that term *x*<sub><i>i</i></sub> is present in a document randomly selected from the ideal answer set

• **Pr**(*x*<sub><i>i</i></sub> | nonrel): probability that term *x*<sub><i>i</i></sub> is present in a document randomly selected from outside the ideal answer set
• Considering document
  \( D = \langle d_1, d_2, \ldots, d_t \rangle \), where \( d_i \) is the
  weight of term \( i \),

  \[
  \log g(x) = \prod_{i=1}^{t} \frac{\Pr(x_i = d_i \mid \text{rel})}{\Pr(x_i = d_i \mid \text{nonrel})} + C
  \]

  where \( \Pr(x_i = d_i \mid \text{rel}) \) is the
  probability that a relevant document
  contains term \( x_i \) (similarly for
  \( \Pr(x_i = d_i \mid \text{nonrel}) \))

• When \( d_i = 0 \) we want the
  contribution of term \( i \) to \( g(x) \) to be 0:

  \[
  \log g(x) = \prod_{i=1}^{t} \log \frac{\Pr(x_i = d_i \mid \text{rel})}{\Pr(x_i = d_i \mid \text{nonrel})} \frac{\Pr(x_i = 0 \mid \text{nonrel})}{\Pr(x_i = 0 \mid \text{rel})} + C
  \]

  =

  \[
  \log g(x) = \prod_{i=1}^{t} \log \frac{p_i(1 \square q_i)}{q_i(1 \square p_i)} + C
  \]
The term relevance weight of term $x_i$ is:

$$tr_i = \log \frac{p_i(1-q_i)}{q_i(1-p_i)} = \log \frac{\Pr(x_i = d_i \mid \text{rel})}{\Pr(x_i = d_i \mid \text{nonrel})} \cdot \frac{\Pr(x_i = 0 \mid \text{nonrel})}{\Pr(x_i = 0 \mid \text{rel})}$$

Weight of term $i$ in document $j$ is:

$$w_{ij} = tf_{ij} \times tr_i$$
Estimation of term occurrence probability

- Given a query, a document collection can be partitioned into a relevant and non-relevant set

- The importance of a term $j$ is its discriminatory power in distinguishing between relevant and nonrelevant documents
• With complete information about the relevant and nonrelevant document sets we can estimate \( p_j \) and \( q_j \):

\[
p_j = \frac{r_j}{R} \quad q_j = \frac{df_j \square r_j}{N \square R}
\]

\[
tr_j = \log \frac{\frac{r_j}{R}}{(df_j \square r_j)/(N \square R)} \frac{1 \square (df_j \square r_j)/(N \square R)}{1 \square r_j/R} = \log \frac{r_j}{R \square r_j} \frac{N \square df_j \square R + r_j}{df_j \square r_j}
\]

• Approximation: \( tr_j = \log \frac{r_j \cdot N}{R \cdot df_j} \)
Term Occurrence Probability

Without Relevance Information

• What do we do because we don’t know \( r_j \)?

\[
q_j = \frac{df_j}{N} : \text{since most docs are nonrelevant}
\]

\[
p_j = 0.5 \text{ (arbitrary)}
\]

\[
tr_j = \log((N/df_j) \cdot 1) : \text{does this remind you of anything?}
\]

• Reminder... Ranking Function

\[
\log g(x) = \sum_{i=1}^{t} \log \frac{p_i(1-q_i)}{q_i(1-p_i)} + C
\]

where,

\[
\begin{align*}
p_i &= \Pr(x_i=d_i|rel) \\
q_i &= \Pr(x_i=d_i|nonrel)
\end{align*}
\]

and \( d_i \) is the weight of term \( i \)