Relevance Feedback

Introduction

- attempt to improve performance by modifying the user query; the new modified query is then resubmitted to the system.
- typically, user examines returned list of documents and marks those which are relevant
- the new query is usually created via:
  - incorporating new terms
  - re-weighting existing terms

‘

Different approaches

- feedback from user used to recalculate weights
- analysis of document set:
  - local analysis (returned set)
  - global analysis (whole document set)

- This feedback allows reformulation of query
- Advantage: User is shielded from task of query reformulation and from the inner details of the comparison algorithm

Feedback in the Vector space model

- assume relevant documents have similarly weighted term vectors
- D_R is the set of relevant documents returned
- D_N is the set of non-relevant documents returned
- C_R relevant documents in whole collection

- Assume C_r is known for a query q
- The best vector for a query to distinguish relevant documents from non-relevant ones is:

- Assume C_r is known for a query q
- The best vector for a query to distinguish relevant documents from non-relevant ones is:

$$\vec{q} = (\frac{1}{|C_r|} \sum_{d_j \in C_r} d_j) - (\frac{1}{N - |C_r|} \sum_{d_j \notin C_r} d_j)$$

- Impossible to generate this query as we do not know C_r
- Can estimate C_r though as we know D_R which is a subset of C_r
- Main approach: Rocchio:

$$\vec{q_{new}} = \alpha \vec{q_{orig}} + \frac{\beta}{|D_r|} \sum_{d_j \in D_R} d_j - \frac{\gamma}{|D_N|} \sum_{d_j \in D_n} d_j$$

α, β, γ are constants which determine:

- importance of feedback
- the relative importance of positive feedback over negative feedback

Variants

- Ide-Regular

$$\vec{q_{new}} = \alpha \vec{q_{old}} + \beta \sum_{d_j \in D_R} d_j - \gamma \sum_{d_j \in D_n} d_j$$

- Ide Dec-Hi (based on assumption that positive feedback is more useful than negative feedback)

$$\vec{q_{new}} = \alpha \vec{q_{old}} + \beta \sum_{d_j \in D_R} d_j - \gamma MAXNR(d_j)$$

where $MAXNR(d_j)$ is the highest ranked non relevant document.

- The use of these feedback mechanisms have shown marked improvement in precision and recall of system
- Salton indicated, in early work on the vector space model, increases in average precision of at least 10%

Evaluation

- recalculate precision-recall for new returned set
- often calculated with respect to returned document set less the set marked by the user

Pseudo-Feedback/Blind Feedback

Local Analysis

- Documents retrieved are examined at query time to determine terms for query expansion
- Typically develop some form of term-term correlation matrix
- To quantify connection between two terms expand query to include terms correlated to the query terms

Association Cluster

- Create matrix M
- can create term x term matrix to represent the level of association between terms
- Usually weighted according to:

$$Mi, j = \frac{freq_{i,j}}{freq_i + freq_j - freq_{i,j}}$$

Query expansion with local analysis

- Can develop an association cluster for each term t_i in the query. For each term $t_i \in q$:
  - choose i^{th}
  - select top N values from row
- For query q, select a cluster for each query term $|q|$ clusters formed
- N is usually small to prevent generation of very large query
- May then take all terms, or those with the highest summed correlation enditemize

Metric Clusters

- association clusters do not take into account position within documents
- metric clusters attempt to overcome this limitation
- Let $dis(ti, tj)$ be the distance between two terms $t_i andt_j$ in the same document.
- if ti and tj are in different documents, then $dis(di, dj) = \infty$
- Can define term-term correlation matrix by:

$$Mi, j = \sum_{t_i, t_j \in D_i} \frac{1}{dis(t_i, t_j)}$$

- Can define clusters as before

Scalar Clusters

- Based on comparing sets of words
- If two terms have similar neighbourhoods there is a high correlation between terms
- Similarity can be based on comparing the two vectors representing the neighbourhoods
- This measure can be used to define term-term correlation matrix and procedure continues as before

Global Analysis

Global Analysis

- based on analysis of whole document collection and not just the returned set
- A similarity matrix is created. The technique used is similar to the method used in the vector space comparison
- Index each term by the documents in which the term is contained
- It is then possible to calculate similarity between the 2 terms by taking some measure of there two vectors - e.g. dot product

To use this to expand a query:

- we map the query to the document-term space
- calculate similarity between query vector and vectors
- associated with query terms
- rank the vectors $\vec{t_i}$ based on similarity
- choose top ranked terms to add to the query

Other issues

- The Rocchio and Ide methods can be used in all the vector based approaches
- Feedback is an implicit component of many of the other IR models (e.g. neural and probabilsitic models)
- Same approaches (with some modifications) in information filtering.

User Feedback

Some problems exist in obtaining user feedback

- Users tend to not give a high degree of feedback
- Users are typically inconsistent in their feedback
- Explicit user feedback does not have to be strictly binary. We can allow a range of values
- Implicit Feedback can also be used. Can make assumptions that a user finds an article useful if:
  - user reads article
  - users spends a certain amount of time reading the article.
  - user saves or prints article.
- These metrics are rarely as trustworthy as explicit feedback