QUERY AND DOCUMENT EXPANSION IN TEXT RETRIEVAL

Clara Isabel Cabezas

University of Maryland College Park

May, 2nd 2000

1.- Definition

2.- Query expansion

a) Different approaches
b) Application example: Ballesteros and Croft
c) Conclusions

3.- Document expansion

a) Document vs query expansion
b) Application: Singhal and Pereira
c) Conclusions

Definition of Query and Document Expansion

The different techniques used to enhance document retrieval by adding new words that will probably appear in documents relevant to the user.

Two main approaches

- Expansion at query time (query expansion)
- Expansion at index time (document expansion)

Why do we need expansion?

- User feedback
- Noisy documents (translation and voice recognition output)

Query Expansion

- **User relevance feedback**

  Selecting a number of terms from the retrieved documents, which have been indexed as relevant by the user. Using these terms as query terms in a new search.

- **Automatic Local Analysis**

  Identifies terms in the retrieved document that are close to the query (synonymity, morphological and derivational variations, terms that frequently co-occur with the query terms, etc.)

- **Automatic Global Analysis**

  Analyses the whole collection is used to create thesaurus-like structures which define term relationships. The user chooses terms from these structures in order to retrieve new documents.

A.-User Relevance Feedback

- Query Expansion
- Term Weighting

Applications:

- Query Expansion and Term Reweighing for the Vector Space Model
- Term Reweighing for the Probabilistic Model

Query expansion and term reweighing for the Vector Model

If the term weight vectors of the documents identified by the user are similar, then we should modify the query vector, so that it also similar to the ones in the relevant documents.

In an ideal situation:

$$\vec{q}_{opt} = \frac{1}{|C_r|} \sum_{\forall \vec{d}_j \in C_r} \vec{d}_j - \frac{1}{N - |C_r|} \sum_{\forall \vec{d}_j \notin C_r} \vec{d}_j$$

Dr = set the relevant docs for the user among retrieved docs

Dn = set the non relevant docs for the user among retrieved docs

Cr = set of relevant docs in the whole collection

|Dr|, |Dn|, |Cr| = number of docs in the sets Dr, Dn, and Cr

α, β, γ = tunning constants

Standard_ Rocchio:

$$\vec{q}_m = \alpha \vec{q} + \frac{\beta}{|D_r|} \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \frac{\gamma}{|D_n|} \sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

Id_Regular:

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

Ide_Dec_Hi:

$$\vec{q}_m = \alpha \vec{q} + \beta \sum_{\forall \vec{d}_j \in D_r} \vec{d}_j - \gamma \; max_{non\text{-}relevant}(\vec{d}_j)$$

Max_non_relevant($\vec{d}$j) = highest ranked non-relevant document.
In original Rocchio, $\alpha = 1$ and Ide $\alpha = \beta = \gamma = 1$

Term Reweighting for the Probabilistic Model

Ranks documents similar to a query according to the probabilistic ranking principle:

$$sim(d_j, q) \; \alpha \; \sum_{i=1}^{t} w_{i,q} \, w_{i,j} \left(\log \frac{P(k_i|R)}{1 - P(k_i|R)} + \log \frac{1 - P(k_i|\overline{R})}{P(k_i|\overline{R})} \right)$$

Where:
P(ki|$\underline{R}$) = probability of the term ki to appear in the set of relevant documents
P(ki|R) = probability of the term ki to appear in the set of non-relevant documents

Since P(ki|R) and P(ki|$\overline{R}$) are unknown,

1) For the initial search:

$$sim_{initial}(d_j, q) = \sum_{i}^{t} w_{i,q}\ w_{i,j}\ \log \frac{N - n_i}{n_i}$$

P(ki|R) = 0.5
P(ki|$\overline{R}$) = ni / N

2) For the feedback series:

$$sim(d_j, q) = \sum_{i=1}^{t} w_{i,q}\ w_{i,j}\ \log \left[\frac{|D_{r,i}|}{|D_r| - |D_{r,i}|} \div \frac{n_i - |D_{r,i}|}{N - |D_r| - (n_i - |D_{r,i}|)} \right]$$

Dr = set of relevant retrieved documents according to the user
Dr,i = set of relevant retrieved documents containing the term ki.
Adjustments 0.5 or ni/N are necessary to avoid unsatisfactory results for small values of | Dr | and | Dr,i |

Advantages of this approach

- The feedback process is directly related to the derivation of new weights for query terms
- This reweighting is optimal, assuming term in independence and binary document indexing.

Disadvantages

- Document term weights are not considered in the feedback series
- Previous query term weight are discarded
- No query expansion is used
- This approach does not work as well as Vector Model Relevance Feedback

B.- Automatic Local Analysis

- Identifies terms in the retrieved documents that are close to the query (synonymity, morphological and derivational variations, terms that frequently co-occur with the query terms, etc.)
- These terms are added to the query for a new search.

Types:

- ***Query expansion through local clustering***

- ***Query expansion through local context analysis***

Query expansion through local clustering

- Finds out terms that appear close to the query terms in documents by structures such as association matrices and use those terms for query expansion.

Type of Clusters:

- Association cluster
- Metric cluster
- Scalar cluster

Given:

V(s) = set of grammatical forms of a word.
e.g.V(s)={write, writes, writing, etc.} where s = write
Dl = *local document set*
Vl = *local vocabulary* (i.e. all distinct words in Dl)
Sl = The set of all distinct words in Vl.

Association Clusters

Terms that co-occur frequently in the same documents are ‘synonymous’.

fsi, j = frequency of stem si in document dj ∈ Dl

Given $\vec{m}$ = (mij) with |Sl| rows and |Dl| columns (where mij = fsi, j) and $\vec{m}$t(transpose of m)

The matrix $\vec{S}$ = $\vec{m}\vec{m}$t, where every element expresses the correlation Cu,v between the stems Su and Sv

$$c_{u,v} = \sum_{d_j \in D_l} f_{s_u,j} \times f_{s_v,j}$$

This correlation factor calculates the absolute frequencies of co-occurrence. The normalized version:

$$s_{u,v} = \frac{c_{u,v}}{c_{u,u} + c_{v,v} - c_{u,v}}$$

Su(n) calculates the set of n largest correlations. It defines a local association cluster that will be used to expand the query terms.

Metric Clusters

- Co-occurrence + distance between terms
- Two terms that occur in the same sentence are more correlated than two terms far from each other.

r(Ki, Kj) = distance between two terms Ki and Kj (number of words in between).
r(Ki, Kj) = ☺, when terms appear in different documents.

$$c_{u,v} = \sum_{k_i \in V(s_u)} \sum_{k_j \in V(s_v)} \frac{1}{r(k_i, k_j)}$$

Where 1/ r(Ki, Kj) = distance between Ki and Kj

The correlation factor is unnormalized. An alternative normalized factor:

$$s_{u,v} = \frac{c_{u,v}}{|V(s_u)| \times |V(s_v)|}$$

$S_{u(n)}$ calculates the set of n largest correlations of a term. This set will define a local correlation cluster that will be used to expand the query.

Scalar Clusters

Based on the indirect or induced relationship idea.

- If two stems have similar neighbors, they are synonymous
- We compare the two correlation vectors for stem v and stem u (their local correlation clusters) with a scalar measure (i.e. cosine of the angle of the two vectors).

$$s_{u,v} = \frac{\vec{s}_u \cdot \vec{s}_v}{|\vec{s}_u| \times |\vec{s}_v|}$$

Where: Su,v = correlation between term u and term v

Su = correlation vectors for stem u

Sv = correlation vectors for stem v

Su(n) calculates the set of n largest correlations of a term. This set will define a local correlation cluster that will be used to expand the query.

Query expansion through Local Context Analysis

- Uses global and local analysis
- Uses noun groups instead of keywords as concepts
- Concepts for query expansion extracted from top retrieved documents
- Uses document passages for determining co-occurrence

Three steps:

1.- Initial top n ranked passages (by breaking top documents in fixed length passages)

2.- Similarity between each concept in the top passages and the whole query

3.- Top ranked concepts added to query

--A weight of 2 is added to each query term.

--A weight is assigned to each added concept. Given by 1 – 0.9 x i/m
(where I = position of the document in the final ranking)

Similarity computation between each concept in top ranked passages and query:

$$sim(q,c) = \prod_{k_i \in q} \left(\delta + \frac{\log(f(c,k_i) \times idf_c)}{\log n} \right)^{idf_i}$$

n = number of top ranked passages

δ = constant parameter to avoid sim(q,c) = 0, usually close to 0.1

Idfi = emphasizes infrequent query terms

f(c,Ki) = association correlation between concept c and query term Ki

$$f(c,k_i) = \sum_{j=1}^{n} pf_{i,j} \times pf_{c,j}$$

Expected that metric correlation gives better results

Query Expansion in Cross-language IR

Ballesteros and Croft 1997

Use query expansion to improve results for Cross-lingual document retrieval

Translation is necessary but lowers performance:

- Machine Translation
- Parallel or Comparable Corpora techniques
- Machine readable dictionary
  - Cheap and uncomplicated
  - Drops 40-60% below monolingual retrieval effectiveness

Causes for bad performance in MRD:

- Out of vocabulary words(e.g. technical terms)
- Addition of extraneous words to the translation
- Bad translation of multiterm phrases

Approaches using query expansion:

- Query expansion before translation
- Query expansion after translation
- Both before and after translation

Their experiment:

Comparison of retrieval using MRD without query expansion, with local feedback, and local context analysis query expansion.

Languages:

- Source Language: English
- Target Language: Spanish

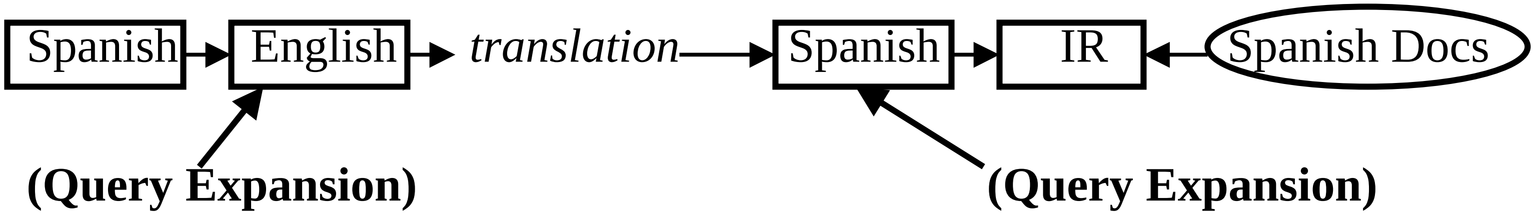
Collection:

- En Norte and San Jose Mercury News(208 and 301 Mb respectively)

IR System:

- INQUERY

Their system:



Pre-translation Query Expansion

(comparison of Local Feedback and Local Context Analysis)

- Collins Spanish-English MRD
- Phrase translation whenever possible. Otherwise word by word

Las relaciones economicas y comerciales entre Mexico y Canada
The economic and (commercial relations) between mexico and canada
Economic(commercial relations) mexico canada Mexico(trade agreement)(trade zone)cuba salinas
[economico equitativo][comercio negocio trafico industria][narracion relato relacion][Mejico Mexico]Canada[Mejico Mexico]

1.- Original
2.- BASE (translation)
3.- LCA expanded BASE
4.- WBW + phrasal translation of LCA expanded BASE

Method	Avg.	% change
MRD	0.0826	
MRD + Phr	0.0826	0.3
MRD + LCA-WBW	0.0969	17.7
MRD + LCA-phr	0.1009	22.7
MRD + Phr+LCA-Phr	0.1053	27.9
LF	0.1099	33.5

--Phrase translation is beneficial

--Still LCA is less effective than LF(LCA more sensitive to wrong phrasal translations)

Post-translation Query Expansion

- Added concepts

  Top-1 got a weight of 1.0

  Each additional one’s weight decreased by 1/100 of the previous one’s weight

  LF added top 20 concepts from the top 50 documents

  LCA added top 100 concepts from the top 20 passages

Economic commercial relations mexico european countries
Comerc narr relat rel econom equit rentabl pai patri camp region tierr mej mex europ
(est un) canada pai europ francie (diversific comerc) mex polit pais alemani rentabl oportum product apoy australi (merc eurp) agricultor bancarrot region (comun econom europ)

1.- Base translation

2.- MRD translation of BASE

3.- 20 post-translation LCA expansion

	MRD	LF	LCA 20
Avg prec	0.0824	0.0916	0.1022
% change		11.3	24.1

- Without bad phase translation factor, LCA seems to perform better than LF

Combined Pre- and Post-translation

- At Post-translation 50 top terms from 20 top passages were added

las relaciones economicas y comerciales entre Mexico y Canada
the economic and (commercial relations) between mexico and canada
economic(commercial relations) mexico canada mexico free-trade canada trade mexican salinas Cuba pact economies barriers
[economico equitativo][comercio negocio trafico industria][narracion relato relacion][Mejico Mexico]Canada[Mejico Mexico][convenio comercial][comercio negocio trafico industria]zona cuba salinas
canada(libr comerci) trat ottaw dosm (acuer paralel)norteamer(est un)(tres pais) import eu (vit econom) comerci (centr econom)(barrer comerc)(increment subit)superpot rel acuerd negoci

1.- Original

2.- BASE + Phr

3.- LCA expanded BASE

4.- WBD + phr translation

5.- LCA expanded translation

Results

	MRD	LF	LCA20-50
Avg prec	0.0823	0.1242	0.1358
% change		51.0	65.0

- Combined method is more effective than Pre- and Post-translation.
- LCA is better at precision (appropriate for Cross-Lingual IR)

Method	Precision	% Monolingual
Monolingual	0.1998	
MRD	0.0823	41.2
Pre-LF	0.1099	55.0
Pre-LCA	0.1139	57.0
Post-LF	0.0916	45.8
Post-LCA	0.1022	51.1
Comb-LF	0.1242	62.2
Comb-LCA	0.1358	68.0

Conclusions

- Machine Readable Dictionary translation to cheap, fast method for CLIR
- The quality of translation affects retrieval
- Poor phrasal translation decreases effectiveness in retrieval
- Both Local Feedback and Local Context Analysis reduce the effects of the poor translations
- LCA gives higher precision (particularly at low recall levels)
- Combined Pre- and Post- LCA expansion gives the best results

    -Reduces translation errors over 45% of MRD

    -From 42% to 68% of monolingual IR

    -Improvement in phrasal translation should help in reducing the gap

Document Expansion

Singhal & Pereira 1999

Document expansion

Use document expansion to improve retrieval results in speech recognition collections

Motivation:

- Speech recognition errors affect retrieval effectiveness (15%-27% worse)

Type of errors:

- Vocabulary mismatch: Words originally in the collection will be deleted(i.e. not be indexed)

  New words will be introduced

- Term-weighting problem: Increasing or decreasing the number of words in the documents will provoke incorrect weight assignment

Some solutions:

- Use of thesaurii. Expensive
- Query expansions
- Document clustering
- Latent Semantic Indexing
- Document expansion

Hypothesis:

In erroneous transcriptions, no certainty that documents are about terms recognized by the ASR system.

Using a comparable corpus (e.g. a side collection from newspaper news), we can:

- Modify Term Weighting (reinforce on-topic words and reduce the importance of off-topic ones)
- Add new on-topic terms (from side-collection).

Their experiment:

TREC SDR track (100 hrs. approx. radio/TV broadcast news)

Manually segmented into 2,866 different stories

23 sentence-length queries (each one has 1-60 relevant docs. in the collection)

Effects of ASR mistakes

Type of mistakes:

- Deletions: a term in the speech fails to be recognized by ASR
- Weight differences: a term in the speech is recognized with the wrong frequency
- Insertion: a term in the speech appears in the ASR output

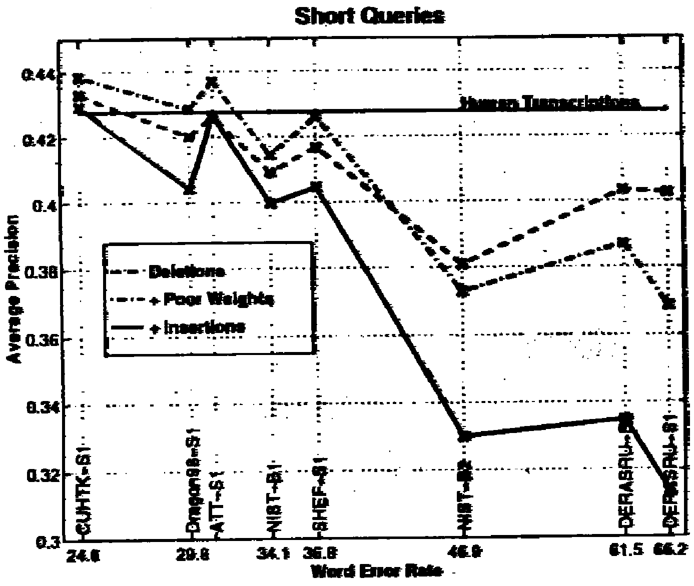
Artificially reproduced these mistakes in the document vectors for human transcriptions to determine the influence of the mistakes in retrieval. Process done incrementally.

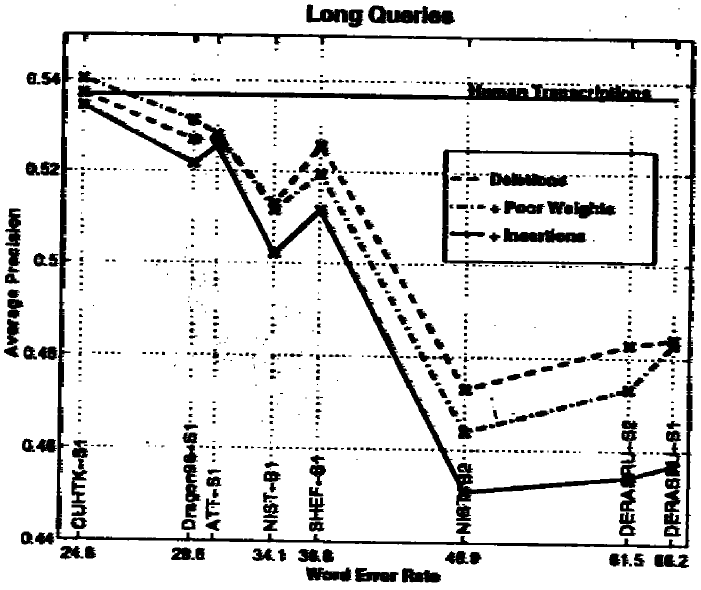
1.- Removal from document vectors of human transcriptions of all term which do not occur in ASR output

2.- Change of weights of the terms in the truncated vectors produce in step one to make them equal to the weights they have ASR output

3.- Insertions to vectors generated from step 2

Results





- Long queries better than short ones(avrg. Prec. For long queries = 0.5369, short queries = 0.4277)
- Loss of retrieval effectiveness smaller in good transcriptions
- Deletions matter (for both short and long queries)
- Weight changes are less important (due to normalization in modern ranking systems)
- Insertions matter for short queries

Document Expansion

Given a noisy document, find nearest neighbor documents (using a side collection) and add words to the former from the latter

ASR collection

TREC-7 SDR automatically transcribed collection

Side collection

North American News corpus (LDC collection of print news)

How comparable do the side collection need to be?

For this experiment: Both corpus were North American news from approximately the same period.

Finding the nearest neighbors:

- ASR collection run as a query against NA News corpus
- Retrieval of the 10 most similar documents

Modification of speech transcription vectors:

Using Rocchio’s formula:

$$\vec{D}_{new} = \alpha \vec{D}_{old} + \frac{\sum_{i=1}^{10} \vec{D}_i}{10}$$

Where:

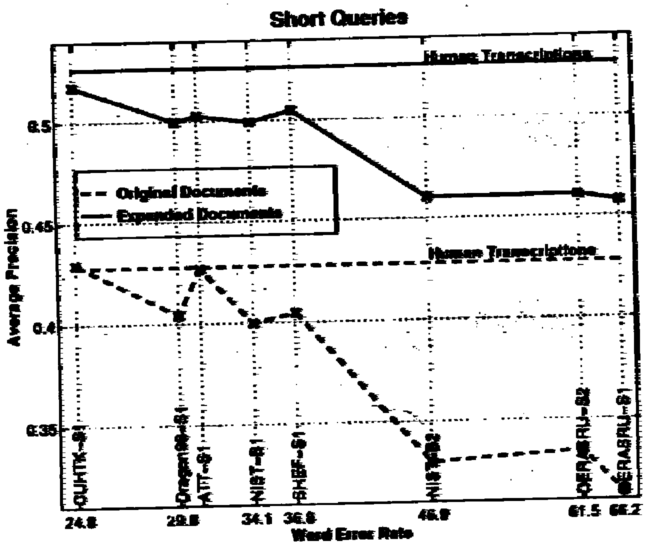
$\vec{D}$new = modified ASR document vector

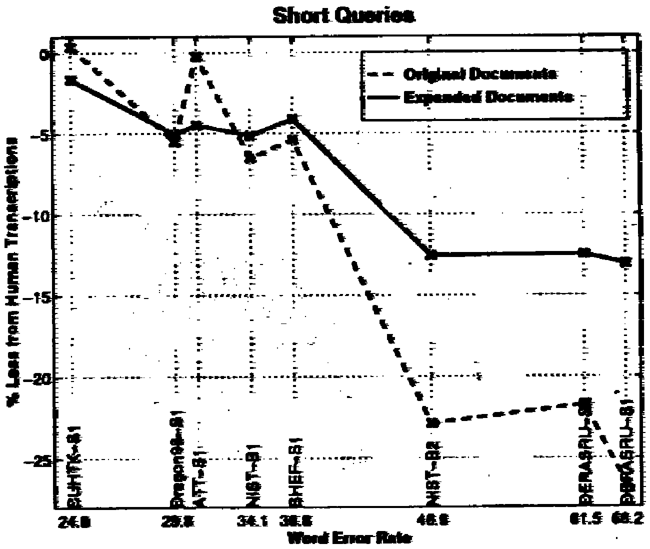
$\vec{D}$old = old document vector

$\vec{D}$i = vector of the ith retrieved document

Addition of n number of words with the highest idfs.

Results for short queries





- Improvement of average precision for short queries for all transcriptions

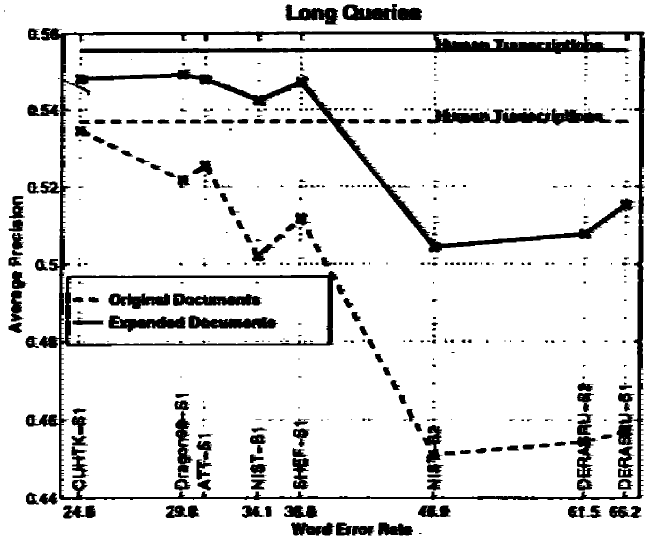
  --23% (0.4277 to 0.5265) improvement for human transcription

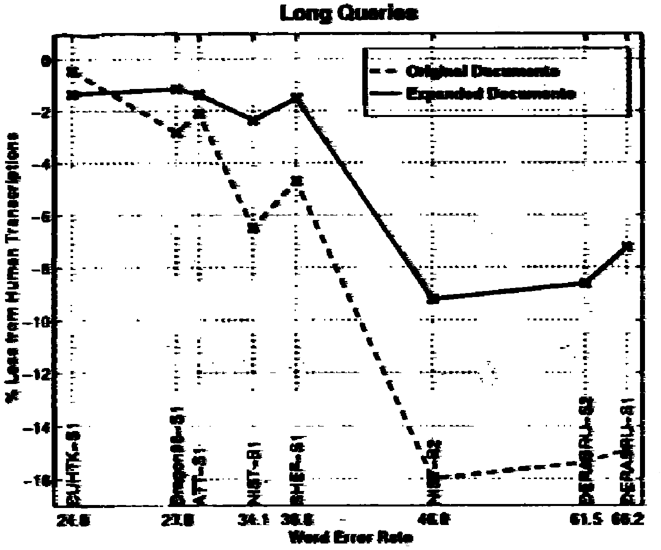
  --Up to 44% for automatic transcriptions – 0.3139 to 0.4576
- Particular beneficial for erroneous transcriptions

  -- CUHTK-S1 does not almost show difference in precision

  -- DERASRU-S1 27%

Results for long queries





- Not as big improvement as for short queries
  --3.5% for human translations
  --12-13% improvement for the worst ASR systems, 3-5% for the best ones
- Some important parameters were decide experimentally:
  10 nearest neighbors
  Short queries → α = 1.0, 10 documents, 100% expansion
  Long queries → α = 1.5 or 2.00, 10 documents, 50-100% expansion
  For both → α = 1.0, 10 documents extracted from side collection, 100% expansion

Conclusions

- Effect of reweighting:

  -- Output ASR documents are considered to contain ‘signal’ and ‘noise’concepts (e.g. in a document about syntax the words ‘lexeme’and ‘China’ respectively.)

  -- The effect of reweighting the concept has the effect of boosting the signal (i.e. give more importance to ‘lexeme’ than to ‘China’ in D1)

  -- This helps both long and short query retrieval and both human and automatic speech transcription.

- Effect of adding new terms:

  -- It helps mainly short queries and bad ASR transcriptions. It does not help or it can hurt long queries, by introducing noise in some cases.

References

- Chapter 5 (Query Operations) in Baeza-Yates and Ribeiro-Neto, *Modern Information Retrieval*.
- A. Singhal and F. Pereira, "Document Expansion for Speech Retrieval," Proc. of SIGIR, Berkeley, 1999
- L. Ballesteros and W. B. Croft, "Resolving Ambiguity for Cross-language Retrieval," in Proc. of ACM SIGIR, Melbourne, Australia, 1998
- L. Ballesteros and W. B. Croft, "Phrasal Translationa and Query Expansion Techniques for Cross-Language Information Retrieval," in Proc. of ACM SIGIR, Philadelphia, 1997.