Thomson Legal and Regulatory at NTCIR-4: Primarily monolingual experiments

Isabelle Moulinier
TLR, Research and Development
isabelle.moulinier@thomson.com

May 27, 2004

Overview

- System overview
- Monolingual experiments in Japanese, Chinese and Korean
  - Creating stopword lists
  - Handling compound terms in Korean
  - My first steps with Pseudo-Relevance Feedback
- Pivot-Language experiments
- Conclusion

System overview

- Research version of a production system
  - Asian languages not in production
- Handled documents in XML
  - Language identified at the collection level
- Indexing is word-based
  - Tokenization and stemming using LinguistX toolkit
- Retrieval model: a cousin of INQUERY
  - Uses structured queries
  - Uses tf-idf for concept scoring

Creating stopword lists

- Using collection information
  - with manual editing (Japanese and Chinese)
  - without manual editing (all languages)
    - 100 or 200 most frequent terms in the collection

- Using query log information
  - without manual editing (all languages)
    - terms appearing in more than 20% of the queries

Main results with stopword experiments

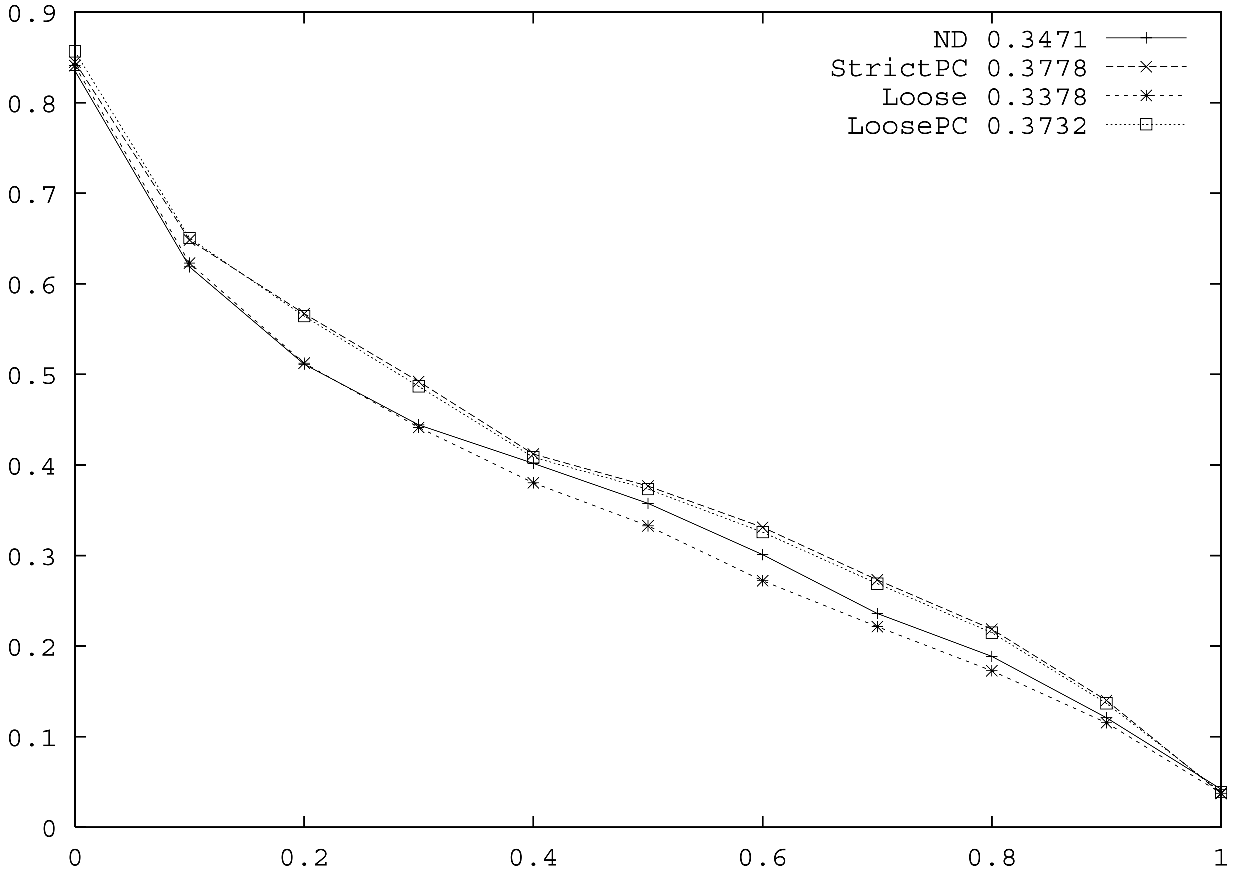
- Using stopword lists usually improves average precision significantly
  - Title only queries contain few stopwords
- Average Precision is not significantly different with various stopword lists
  - Typical stopwords appear in all lists
  - There is a query per query difference
- Key is balance between stopwords and concepts
  - Full queries contain strong concepts thanks to concept fields

Handling Korean compounds

- We use a stemmer to identify compound parts
  - Example: 홈런경쟁에서 stems to 홈런#경쟁
- We think of compounds as equivalent to phrases
- Our approach
  - Index compounds and their parts
  - Use different proximity structures

	No Partial Credit	Partial Credit
Strict (ND)	홈런#경쟁	홈런#경쟁$_w$ 홈런$_{w_1}$ 경쟁$_{w_1}$
Loose	NPHR(홈런 경쟁)	NPHR(홈런 경쟁)$_w$ 홈런$_{w_1}$ 경쟁$_{w_1}$

Results with Korean compounds



- Partial credit is helpful

- Key is on good compound recognition

First steps with pseudo-relevance feedback

- Query expansion using PRF
  - Terms are selected using Rocchio's formula and added to the original query

$$sw = \frac{\beta}{|R|} \sum_{d \in R} (ntf * nidf) - \frac{\gamma}{|\overline{R}|} \sum_{d \in \overline{R}} (ntf * nidf)$$

- Parameter tuning using NTCIR-3 data
  - select 20 terms
  - select the 5 first documents as relevant
  - select the last 20 documents as irrelevant
  - $\beta = \gamma = 1$

PRF results

- Some improvement over base runs but no statistical difference
- Large query variations

	$\Delta > 10\%$ (+/-)	$\Delta > 20\%$ (+/-)	$\Delta > 40\%$ (+/-)
tlrrd-tdnc-01	24 (14/10)	12 (7/5)	2 (0/2)
tlrrd-t-02	45 (18/27)	35(13/22)	21 (6/15)
tlrrd-t-03	39 (19/20)	27 (14/13)	16 (7/9)
tlrrd-dn-04	27 (17/10)	18 (13/5)	4 (3/1)

- Noticeable improvement in precision at 5 documents
- Key is finding good documents in the original search

Pivot-language IR using Web resources

- Goal: Assess how well (poorly) pivot-language translation using Web resources would work
- Korean-English-Japanese
  - sentence translation using Babelfish
- Chinese-English-Japanese
  - word-based translation using Chinese-English dictionary and Babelfish from English to Japanese
- Outcome: Pivot-language IR uses Web resources works POORLY.

Conclusion

- Below average performance for our official runs
  - word-based indexing, especially tokenization
- Monolingual results
  - Stopwords are useful, independent of how they are created
  - Partial credit useful for searching compounds
  - Below expectation results for PRF
- Pivot-language results
  - The “dumb” approach does not work
  - Good news for more research