InTEXT Automatic Query Enhancement in TREC-5

by

Richard Jones, Mark Burnett, and Lewis Pape
InTEXT Systems,
P.O Box 310 Deakin West
ACT 2600 Australia

100032.1075@compuserve.com

A. Background

InTEXT Systems is a subsidiary of the CP Software Group, whose headquarters are in Folsom, California. InTEXT is a leader in the provision of advanced tools and end-user products that can best be described as doing 'smart things with text'. The InTEXT Research and Development group, based in Canberra Australia, has been in existence for 11 years. We develop leading edge technology in areas including text retrieval, filtering, indexing, summarisation and thematic clustering, using the InTEXT Heuristic Learning Architecture.

The InTEXT Systems R and D group has participated in two previous TRECs. We took part in TREC-4 and also in TREC-1, when the team formed the Centre for Electronic Document Research (CEDR), in conjunction with CITRI.

B. Objectives of the Experiment

The experiments INTXA, and INTXM described in this paper were carried out in the filtering stream of TREC-5. The principal objective of these experiments was to determine the effectiveness of the InTEXT Precision indexing tool to automatically improve queries. The experiments were run with the InTEXT Object Router tool-kit, and a second objective was to determine the effectiveness of Object Router in the TREC filtering environment.

In particular we wanted to see if InTEXT Precision could be used to take an initial query, manually created with minimal effort and transform it into an improved query, taking automatically selected words and phrases from a group of documents that were deemed relevant to the original query

The motivation behind the experiment was that in a filtering environment, queries are long-lived objects and so the return on investment in improving them will be greater than for an ad-hoc query. However, end users do not enjoy the intricacies of query building. In addition, a time specific cohort of documents relevant to the query will share a number of concepts other than those that caused them to be selected. These terms can be useful in

improving a query. However, some terms will be ephemeral (eg references to the Exxon Valdez in connection with environmental disasters), and a different document cohort from a different time will have an overlapping, but not identical set of concepts that distinguish them. Therefore a query needs to be modified with time, but if users are reluctant to create a good initial query, they will be doubly reluctant to keep it up to date.

C. The Experiment

The initial 50 queries were built manually using the information contained in the original TREC queries as a base (except in those taken from TREC-4, where we used our own queries). Minimal effort was used in building the queries. This set formed the experiment INTXM.

All the training documents judged relevant to the queries in earlier TREC experiments were extracted and these were run through all the INTXM queries in InTEXT Object Router. For each query the 15 documents judged most relevant by Object Router were selected. Those paragraphs containing search terms were extracted from each document and concatenated to form a composite document.

These 50 composite documents were run through InTEXT Precision to extract a set of key words, ordered by their automatically generated weights. These measure their contribution to the information content of the document.

The keywords were then used to create a second set of queries, following the rules defined in Appendix 1. Note that some existing terms were removed or modified by the rules, as well as new ones being added. For an example of a pair of queries for the same topic, see Appendix 2. These queries formed the experimental set INTXA.

Both sets of queries were dispatched to the TREC organisers. The test documents were run through the 100 queries from the combined experiment sets, and the selections made for evaluation under the three utility functions defined in the experiment.

Object Router derives a relevance score for each document, based on the criteria outlined below. The method chosen for document selection was to take a cut-off at a percentage of the highest relevance score achieved for a given query. The percentages selected were: 66%, 33% and 16% from Precision Oriented to Recall oriented.

D. The InTEXT Tool-kits

InTEXT Object Router provides a measurement of the relevance of a document stream to a number of queries. The queries or filters may either be expressed in plain English, or in a structured form, as in Appendix 2, together with a threshold score in the range 1-100 defining the relevance that a document must reach to be passed to the filter owner. Relevance is measured by a set of heuristics, which include presence of the number of terms, and their distribution through the document, with more value being assigned to

closely grouped terms. The software, which was used in an earlier version in TREC-1 has two interesting features:

- the queries are indexed since they are the long lived objects, but the documents are not, as they are ephemeral. As a result the Object Router can efficiently support many active filters, in excess of 100,000;

- the query terms are automatically weighted, depending on their ability in the context of the particular query to act as a good discriminator. The weighting is carried out dynamically on the document stream, so the term weights did vary during the experimental runs.

InTEXT Precision was described in TREC-4. It analyses document content, using techniques analogous to skim reading to identify, rank and categorise keywords and phrases, referred to as Keys. Queries were automatically generated from the list of Keys, depending on their rank and type (for example, proper noun, acronym etc).

Precision also generates 'skinny documents' to support full text indexing whilst effectively generating a stop word list for each document and reducing the number of words to be indexed by a factor of 10. This was the basis of the TREC-4 experiment

E. Results of the Experiments

The basic question we were interested in answering in the experiment was whether INTXA earned more money (or lost less) than INTXM.

The mean results per query using the Pool utility calculation were as follows:

	Precision-Oriented	**Balanced**	**Recall-Oriented**
INTXA	-11	-32	-48
INTXM	-281	-451	-405

So on the base figure from the utility function, the modified queries performed considerably better than the unmodified one, even though neither were brilliant.

Turning to precision and recall across the 45 queries with relevant documents, the official results are:

INTXM	**Precision**	**Balanced**	**Recall**
Total Retrieved	5512	27520	30002
Total Relevant	5808	5808	5808
Relevant Retrieved	611	2322	2438

INTXA	**Precision**	**Balanced**	**Recall**
Total Retrieved	393	3126	9567
Total Relevant	5808	5808	5808
Relevant Retrieved	149	757	1787

INTXM	**Precision**	**Balanced**	**Recall**
Mean Precision	0.3308	0.1679	0.1654
Mean Recall	0.1068	0.4212	0.4338

INTXA	**Precision**	**Balanced**	**Recall**
Mean Precision	0.3331	0.2326	0.1913
Mean Recall	0.0793	0.1894	0.3295

On these statistics, the improvement is not at all clear-cut. though see comment below on improvment on precision figures from above data.

F. Comments

The results are encouraging in the following regard:

- the automatically generated terms looked sensible - see Appendix 2 for an example;
- the automatically generated queries would be particularly useful in an environment demanding high precision. For example, in the Precision-Oriented automatic run 149 out of 393 documents routed were relevant, compared to 611 out of 5512 for the manual queries.

Why the results were not better:

- the training documents from which the modified queries were built were very different from those used in the experiment, in time and in source. The mechanism will work better in an environment where there is some continuity across the data.
- Some of the rules for query modification were too aggressive, in replacing existing terms sometimes with terms that were ephemeral;
- it is clear that the cut-off point decisions were not optimal.

G. Conclusion

We are satisfied that the approach has the potential to gives worth-while results, but that work is needed to improve strategies as described in the comments above.

Appendix 1 - Query Generation/ modification Procedure

A query is made up of terms, approximating to a concept. A term is made up of a number of tokens, which are alternatives. A token may be a word, a phrase, or words collocated in paragraph. In the Object Router, a query is limited to 20 terms. See specific query syntax in Appendix 2.

The original query is modified by the addition of new tokens either into existing terms, or as new terms, and by the removal of original tokens using the rules described below. New tokens are selected in turn from the Keys generated by Precision, starting with the highest scored. For comparison purposes, all Keys and tokens are truncated with the same stemming algorithm.

When the Keys are combined, all proper noun terms with only one token are combined into one term, and the other single token terms are also combined. Then if necessary, tokens are successively removed starting with the lowest ranked by Precision, until the limit is reached.

Rules for incorporating Keys

1. Proper Nouns
- Add as phrase, not paragraph collocation
- add as a new term, unless linked to another token by a shared word
- Accept 1, 2, or 3 word phrases
- match acronyms to proper compound nouns, using InTEXT heuristics

2. Non Proper Nouns
- Add phrase as 'same paragraph' collocation.
- Do not add single words, phrases counted after elimination of noise words
- Allow insertion of 3 word phrases if in top 10 Keys.
- Do not replace two word phrase by three word phrase
- If two 3 word phrases have two words in common, replace by the two word sub-phrase

3. Token addition/ replacement/ deletion
- If A B exists, add any two word phrase containing A and B as extra token
- Delete all tokens not in the keyword list, unless its component words all exist
- If A or B exists as a single word token and A B occurs above A or B in the keyword list then replace by A B. Else add as new term.
- If term choice in token addition, add to shortest term
- do not add tokens to mandatory terms

4. Term addition
- Add 2 term phrases or proper nouns with no current matches as new terms.

Appendix 2 - Example Queries

+ Mandatory term
// same paragraph
* stemmed

INTXM Query 142

1. +grain, animal feed*, corn, rice, wheat
2. export subsid*, export restitution, import quota*, farm trade barrier*
3. price support*, farm subsid*, agricultural support*
4. self sufficiency, embargo, import dependence, high quality food supply, farm interests
5. foreign trade, trade negotiations, agricultural policy
6. Common Agricultural Policy, CAP, Ministry//Agriculture, Zenchu, Liberal Democratic Party, LDP, Export Enhancement, food importing

InTXA Query 142

1. +grain, corn, rice, wheat
2. export subsid*, export restitution, import quota*, export enhance*
3. price support*, farm subsid*, agricultur* support*, low*//price*
4. self sufficiency, embargo, import dependence, farm interests, American//farm*, Rice//farm*, farm//product*
5. foreign trade, trade negotiations, agricultural policy
6. Ministry//Agriculture
7. food import*, import*//quota*
8. import*//rice, rice//miller*, rice//market*
9. export subsidy, wheat//export*
10. Japan*//rice, Japan*//beef

INTXM Query 189

1. +murder*, homicide*, second degree
2. motive*, reasons, jealous*, relation*, partner, family, domestic

INTXA Query189

1. +murder*, second degree
2. motive*, family
3. capital//murder, commit//murder, first-degree//murder, second-degree//murder, open//murder, accused//murder//conspiracy, murder//plot, murder//charges
4. Murder Conspiracy Trial, Appeals Court
5. famil*//member*, famil*//reunion, famil*//honor, family//problem
6. death//penalty, alleged//plot, racial//motive, defense//attorney