# Combining First and Second Order Features in the TREC 2003 Robust Track

Endre Boros, Paul B. Kantor and David J. Neu

November 18, 2003

## Abstract

This year at TREC 2003 we participated in the robust track and investigated the use of very simple retrieval rules based on convex combinations of similarity measures based on first and second order features.

## 1 Introduction

In the robust track, systems attempt to retrieve documents relevant to 100 different information needs, using only the text which is provided in a short descriptive passage known as a topic. The systems submit a list of up to 1000 documents which they attempt to rank by their relevance to the information need.

A generally accepted tenet in information retrieval is that the more topic terms that appear in a document, the more likely that document is to be relevant. It is also widely agreed that the co-occurrence of topic terms is also a good indication of relevance.

We investigated the use of very simple retrieval rules based on convex combinations of similarity measures based on first and second order features, where first order features were terms in the topic and second order features were features designed to capture information about term co-occurrence.

## 2 Approach

The topics in this year's robust track consisted of title, description and narrative sections. Participants were required to submit at least one run which only utilized the description section. All runs we submitted only utilized the description section.

As mentioned in §1 our retrieval rule is based on two different types of features. First-order features are simply the non-stopword terms appearing in the topic description and the first-order topic feature vector for a topic or document is a Boolean vector in which the $i^{th}$ component is 1 if the text contains the $i^{th}$ first order feature and 0 otherwise. The SMART stopword list was utilized.

Second-order features are term pairs which occur within $w$ terms of each other in the topic description prior to the removal of stopwords, and the second-order feature vector for a topic is a vector in which the $i^{th}$ component is the minimum distance between the pair of terms which comprise the $i^{th}$ second order feature in the topic description.

As an example, of second-order feature construction, consider the string, "The focus of the next conference is Boolean functions.". The terms "the", "of", "next" and "is" are stopwords, so the list of non-stopword terms is [ _, "focus", _, _, _, "conference", _, "Boolean", "functions"] . The distance between the non-empty term pairs is shown below:

| Pair | Distance |
|---|---|
| conference, focus | 4 |
| boolean, focus | 6 |
| focus, functions | 7 |
| boolean, conference | 2 |
| conference, functions | 3 |
| boolean, functions | 1 |

So using $w = 3$, the list of the second-order features

would be: [ ("boolean", "conference"), ("conference", "functions"), ("boolean", "functions") ].

As can be seen, we have decided to utilize a purely Boolean model which only captures whether a term, or term pair appears in a document or not, thereby ignoring all term frequency information.

For each document $d$, the score for a given topic is

$$\sigma(d, w, \lambda) = \lambda\phi(d) + (1 - \lambda)\psi(d, w)$$

where the first order similarity measure, $\phi$ is the cosine of the angle between the first order topic feature vector and the first order document feature vector, and the second-order similarity measure $\psi$ is the cosine of the angle between the second order topic feature vector and the second order document feature vector. That is, the score, $\sigma$ is the convex combination (weighted average) of the first-order and second-order similarity measures. We submitted five runs with $\lambda \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$, corresponding to different weightings of the first and second order similarity measures. In all submitted runs, $w = 3$ was used.

# 3 Results

Analysis our performance showed that our scores did not meet expectations. We did attain the median number of relevant retrieved documents at 10, in about one-quarter of the topics, and exceeded it in about 5 percent of the topics, for all our runs. A more detailed comparison between our performance and the median performance is provided below.

| $\lambda$ | Measure | $\geq$ Median | $>$ Median |
|---|---|---|---|
| 0.0 | Rel. Ret. @ 10 | 23 | 3 |
| 0.0 | Avg. Precision | 5 | 5 |
| 0.25 | Rel. Ret. @ 10 | 25 | 3 |
| 0.25 | Avg. Precision | 4 | 4 |
| 0.5 | Rel. Ret. @ 10 | 23 | 5 |
| 0.5 | Avg. Precision | 5 | 5 |
| 0.75 | Rel. Ret. @ 10 | 27 | 5 |
| 0.75 | Avg. Precision | 4 | 4 |
| 1.0 | Rel. Ret. @ 10 | 25 | 8 |
| 1.0 | Avg. Precision | 4 | 3 |

The following two table demonstrated that there was substantial overlap in the topics that performed above the median for the number of relevant documents retrieved at 10 and average precision measures, thereby providing some evidence that $\lambda$ need not be selected on a per topic basis.

| $\lambda$ | Topics in which Rel. Ret. @ 10 Exceeded the Median |
|---|---|
| 0.0, 0.25 | 303, 608, 618 |
| 0.5, 0.75 | 303, 347, 379, 608, 618 |
| 1.0 | 303, 330, 347, 379, 409, 612 618, 628 |

| $\lambda$ | Topics in which Avg. Precision Exceeded the Median |
|---|---|
| 0.0 | 303, 416, 608, 618, 627 |
| 0.25 | 303, 608, 618, 627 |
| 0.5 | 303, 389, 608, 618, 627 |
| 0.75 | 303, 389, 608, 618 |
| 1.0 | 379, 608, 618 |

Finally, an analysis of the detailed results indicates that performance on the new topics was notably better than on the old topics and that performance measures improved slightly as $\lambda$ increased. This later observation indicates that the use of co-occurrence information weakened rather than then improved our performance, which was contrary to expectations.

# 4 Conclusion

Even for such a simple model, our robust track runs performed below expectations. However, the fact that performance on all measures increased slightly with $\lambda$ seems, to indicate that the method could be improved by tuning $\lambda$. In addition, we suspect that utilization of a purely Boolean model and using a relatively small value of $w$ may have negatively impacted performance.

Future research will involve investigation of the impact of varying $w$ as well as the incorporation of term frequency information into our model.

## References


[1] William B. Frakes and Ricardo Baeza-Yates, editors. *Information Retrieval: Data Structures and Algorithms*. Prentice-Hall PTR, 1992.

[2] Elke Mittendorf, Bojidar Mateev, and Peter Schauble. Using the co-occurrence of words for retrieval weighting. *Information Retrieval*, 3(3):243–251, 2000.

[3] C.J. van Rijsbergen. A theoretical basis for use of co-occurrence data in information retrieval. *Journal of Documentation*, 33(2):106–119, 1977.