Using a Reference Corpus as a User Model for Focused Information Retrieval

Gilad Mishne Maarten de Rijke Valentin Jijkoun

Informatics Institute, University of Amsterdam
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands
gilad,mdr,jijkoun@science.uva.nl

ABSTRACT

We propose a method for ranking short information nuggets extracted from a text corpus, using another, reliable reference corpus as a user model. We argue that the availability and usage of such additional corpora is common in a number of IR tasks, and apply the method to answering a form of definition questions. The proposed ranking method makes a substantial improvement in the performance of our system.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing—*linguistic processing*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering, search process*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*question-answering (fact retrieval) systems*; I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems; I.2.7 [**Artificial Intelligence**]: Natural Language Processing

General Terms

Information Retrieval

Keywords

Question Answering, Information Retrieval

1. INTRODUCTION

The area of Question Answering (QA) is at the focus of a lot of research interest lately, both in the Information Retrieval (IR) community and among Computational Linguists. It is seen as one of the few applications to successfully combine techniques from Natural Language Processing and IR. The QA track at the annual Text REtrieval Conferences (TREC, [20]) has become an important factor in shaping and giving direction to QA research. Introduced in 1999, this track attracts a significant number of participants each year, and provides a focal point for much modern QA research.

When the QA track at TREC was introduced, it focused on so-called "factoid" questions (typically having a short named entity as an answer) such as *How many people live in Tokyo?* or *When is the Tulip Festival in Michigan?*. As the track evolved, it was argued that this type of questions does not accurately model the needs of real users of QA technology. In addition to named entities as answers, users often search for definitions of concepts, or for summaries of important information about them. As a result, in 2003 TREC introduced *definition questions*—questions for which the answer is not a single named entity, but a list of *information nuggets* [19].

In the TREC 2004 QA track this was taken a step further. The questions were now clustered in small groups, organized around the same topic. For example, the topic *Concorde* included questions such as *How many seats are in the cabin of a Concorde?* and *What airlines have Concordes in their fleets?*. Finally, for every topic, the track guidelines required participants to supply "additional important information found in the corpus about the target, that was not explicitly asked." This last requirement has been dubbed *"other"* questions [20]. In our view, the task presented at the TREC 2004 QA track, and the introduction of the "other" questions makes a big step towards more realistic user scenarios. According to our own analysis of web query logs, users tend to ask much more "knowledge gathering" questions than factoid questions about specific facts.[1]

This new type of "other" questions puts more emphasis on the *user* aspect in the QA process—an issue that has mostly been neglected in the QA community. The TREC criteria for what is a *good answer* to a given question has so far been rather vague, but QA systems dealt with this vagueness fairly effectively for factoid questions. With the "other" questions, where systems are required to return only *important* information, there is an implicitly assumed user model that can discriminate between important and unimportant facts about a topic. For example, for the topic *Clinton*, his birthday might be considered important, while the day of the week when he left Mexico probably is not. In order to give reasonable responses to "other" questions, a QA system needs to model such preferences.

[1]The analysis of this data is preliminary and will be published elsewhere as soon as it is completed.

DIR 2005, 5th Dutch-Belgian Information Retrieval Workshop.

We present an approach for answering "other" questions using an explicit user model. We describe a method for gathering important facts about an entity from a collection of documents and for ranking the facts with respect to their importance for the user. We show that our ranking improves over plain retrieval of facts from the corpus. The core idea of our method is to estimate the importance of facts found in the target collection by using external "reference" corpora, high-quality sources of information that model a user's ability to distinguish between important and unimportant facts. The proposed method is our first step towards user-oriented QA, and further refinements of the underlying techniques are needed. We identify additional areas where this method is or may be helpful, and discuss its strengths, weaknesses and directions for further research.

The rest of the paper is organized as follows. In Section 2 we survey related work regarding answering definition questions, and about using high-quality external sources. Next, in Section 3 we describe the details of the re-ranking method. Our experiments and results follow in Section 4, and we wrap up with conclusions in Section 5.

2. RELATED WORK

Our approach to ranking snippets extracted from a given document collection is based on "double" ranking: a regular IR ordering by decreasing relevance to the query, followed by re-ranking based on a comparison of those snippets with information mined from a "reference" corpus. Such "double" ranking and re-ranking schemes have been used widely in IR. E.g., in their Maximal Marginal Relevance criterion Carbonell and Goldstein [4] strive to reduce redundancy while maintaining query relevance in selecting appropriate passages for text summarization. Similar ideas have been widely used in work carried out at TREC's novelty track [17], where two things have to be done: relevant sentences should be extracted from a list of relevant documents, and from the resulting list of sentences only the ones contributing novel information should be retained. Kamps [10] and Van Hage et al. [18] use two-step ranking procedures in which the list of documents output by a retrieval engine is re-ranked based on hierarchical relations of relevant metadata concepts in a thesaurus or ontology, respectively.

We are not the first to be using electronic encyclopedias in open domain QA. Kupiec's Murax [11] was probably the first modern open domain QA system, combining IR techniques with shallow natural language processing to answer factoid questions against an electronic encyclopedia. More generally, many teams participating in the TREC QA track use resources other than the corpus against which the questions need to be answered. E.g., at TREC 2002, the University of Waterloo's QA system consulted the web as well as locally developed corpora and knowledge bases with answers to questions of frequently occurring types [6]; IBM's usage of the CYC knowledge base provides another example [5].

Our re-ranking mechanism is related to BBN's use of so-called *question profiles* for re-ranking candidate answers to definitional questions at TREC 2003 [21]. Question profiles are vectors of word frequencies generated from existing definitions of the question target in electronic dictionaries, as well as from biography collections and search engine results.

The work in this paper is also related to the so-called *answer projection* task that data-intensive QA system often face: given an answer to a question (possibly obtained from an outside source), find supporting documents in a given collection for it [15]. Phrased this way, the task resembles a known-item search task. Accordingly, answer projection has been addressed using the kind of high precision retrieval models that have typically been employed for known item search, such as specific Okapi settings [3], passage retrieval, and combinations of heuristics [14].

The use of external, high-quality data sources in IR is not limited to the QA setting. Some examples of IR tasks in which a reference corpus of some kind has been or is being used are *filtering*, *spoken document retrieval (SDR)*, and *web retrieval*. Filtering, which was evaluated at TREC for a large number of years, relies on the availability of standing information needs, possibly with a reference corpus of documents known to be relevant to the information need. In SDR, the corpus is usually noisy (literally), containing incomplete documents; parallel "clean" corpora are often used in this case to expand the noisy documents or the query and improve retrieval significantly [9].

In the popular area of *web retrieval*, some search engines, such as Yahoo! and Google (to some extent), maintain a human-generated catalog of internet sites, in addition to the index of crawled data from the internet. We are not aware of published research on using this catalog to improve retrieval, but this seems a viable option. Even if we take as an example "reference" corpus the relatively small English edition of Wikipedia (`http://en.wikipedia.org`), an open-content encyclopedia, rather than the large human-generated indexes, there are many web queries that can benefit from using it. In query logs released by AltaVista [2], 13% of about 7M queries have an entry in Wikipedia (this was checked without removing stopwords and with no morphological normalization, which will most likely increase the percentage further).

3. RANKING WITH A USER MODEL

In this section we provide the details of our method for extracting, ranking, and re-ranking information nuggets from a corpus. In a nutshell, after identifying a suitable "reference" corpus for our domain (our user model), we first use IR and NLP methods to identify information nuggets—short excerpts of text—related to the topic, both from the given document collection and from the "reference" corpus. Then, we use sentence-similarity metrics to rank the nuggets from the collection: the facts similar to those found in the "reference corpus" are considered more important and ranked higher.

3.1 Target Corpus and Reference Corpus

In the TREC QA task, answers to questions (including "other" questions) must be found in a given text corpus. In recent years, this corpus has been a part of the AQUAINT corpus, containing more than 1 million newswire documents, and a total of 3.1GB of text. In our experiments this corpus is used as the *target* corpus, where important information nuggets have to be located. The corpus is unstructured: we do not know beforehand which articles or passages contain "important" information about a topic.

The "reference" corpus to be used should be a relatively small, high-quality collection of documents, which is catalogued in a way that facilitates selecting documents which contain important information for a given topic. Typical corpora that can be used for such reference purposes are en-

cyclopedias (e.g., biography pages from `http://biography.com`) and various knowledge bases (e.g., the Internet Movie Database `http://www.imdb.com`). Since TREC QA is an open domain task, we used the English edition of Wikipedia (`http://en.wikipedia.org`), an open domain encyclopedia. The version we used contained 768,000 entries (including placeholders and disambiguation entries), for a total of 900 MB of text.

3.2 Mining Facts from the Target Corpus

When answering an "other" question for a given topic, we use IR to locate documents containing information about the topic, and then split the sentences from the retrieved documents into more easily "digestable" shorter nuggets.

Retrieval

First, from the target collection we retrieve the top 20 documents containing the topic as a phrase, using a traditional vector space model for the retrieval. Our collection is composed of news articles with headlines. Since an occurrence of a topic in a headline can be very indicative of the document's importance for the topic, we indexed the headlines and the article bodies separately, and calculate the retrieval score as a combination of the different representations; this is a common technique for semi-structured IR [16].

Extraction

Since a response to an "other" question is a list of short nuggets, we have to split the retrieved documents into separate facts. This raises several problems. First, we observed a notorious use of referential NPs: even in highly focused documents the topic is introduced initially, and then referred to with pronouns or definite NPs (e.g., "*PRESIDENT CLINTON arrived today at the ... HE will leave to Mexico on Monday*"). We therefore resolve pronouns in the documents using a simple anaphora resolution module described in [1]. Then, we extract all sentences which contain the topic (either originally or after the resolution); this is a natural way to restrict our attention to document sections which potentially include facts about the entity.

Still, the sentences are often too long to be presented as nuggets. Moreover, as the next step of our method involves comparison of nuggets, we need to keep them atomic, i.e., as short as possible. We observed that most facts in the extracted sentences could be described with simple predicates (e.g. "[President Clinton] *will leave* to Mexico"). We therefore parse the sentences with Minipar—a wide-coverage dependency parser [12]—and consider as a *fact nugget* every predicate (usually, a verb) with all its arguments and modifiers. Table 1(a) shows an example for the topic *Cassini space probe*.

Finally, every extracted fact is given a *prior importance estimation*: the retrieval score of the document from which the fact was extracted.

3.3 Mining Facts from the Reference Corpus

In order to obtain a list of "good" facts for a given topic, we now repeat the fact extraction stage, with slight modifications, for the reference corpus. First, we extract a high-quality document (i.e., an encyclopedia entry) for the topic. We then apply the anaphora resolution and sentence splitting methods described in the previous section. Next, we assign *importance* to each fact, based on layout cues in the document, such as proximity to the beginning of the entry. These heuristics are based on the fact that in encyclopedia entries, important information is typically given first, data in tables is usually significant, and so on. An example of facts extracted from an encyclopedia entry is given in Table 1(b).

3.4 Estimating Importance of Facts

At this stage, we have two lists of nuggets: facts from the target corpus, with prior importance estimation, and reliable facts from the reference corpus, each with its importance value. To refine the importance estimation for the target facts, we calculate sentence-level similarity between the target and reference nugget lists: we exhaustively compare each target fact to each fact from the reference corpus. We experimented with two types of sentence-level similarity measures: lexical and semantic.

We measure lexical similarity by determining the word overlap between the sentences, using metrics such as Jaccard [8] to normalize over the sentence lengths. Prior to the comparison we use standard stemming and stopword removal on both sentences to increase the morphological uniformity. As to semantic similarity between sentences, we use linguistically motivated techniques to find similarities also between sentences which do not match on the surface level. We use two types of metrics; the first is the total WordNet distance of words appearing in the sentences, based on methods described in [7]. Alternatively, we use similarity scores between pairs of words derived from proximities and co-occurrence in large corpora, described in [13], and sum the total proximity measure for the words in the two segments.

In the experiments described below we used the lexical similarity with Jaccard metric. Later we found that co-occurrence-based measures seem to give better estimates of sentence similarity. A careful evaluation of different measures for this task is in our future plans.

Let $\{t_i\}$ denote the list of facts extracted from the target corpus and $\{r_j\}$ the reliable facts from the reference corpus. We denote the similarity between two facts as $\mathrm{sim}(t_i, r_j)$, the prior importance estimation of a target fact as $I_{pr}(t_i)$ and the importance of the reliable fact as $I(r_j)$. Then the updated, posterior importance estimation of a target fact is calculated as follows:

$$I_{post}(t_i) = I_{pr}(t_i) \cdot \max_j \left(I(r_j) \cdot \mathrm{sim}(t_i, r_j) \right).$$

We sort the target facts by decreasing posterior importance and present the top N as the key facts about the topic.

3.5 Removing Redundant Facts

At the TREC 2004 QA track, each "other" question was asked after a sequence of factoid questions, all about a given topic. Therefore, an additional requirement was set on the response to the "other" question: the retrieved facts should not duplicate the information conveyed by (answers to) the factoid questions.

To avoid such duplication, we performed another filtering step: from the ranked list, we omit nuggets that are *similar* to other nuggets higher in the ranking, or to one of the factoid questions together with its answer (as found by our factoid-QA system). We use the same sentence-level similarity measure $\mathrm{sim}(\cdot, \cdot)$ as for the posterior importance estimation.

Document text	The Cassini space probe , due to be launched from Cape Canaveral in Florida of the United States at dawn , is carrying 33 kg of plutonium needed to power A rocket's seven-year journey to Venus and Saturn . Local mass media quoted opponents of Cassini as saying at the weekend that the mission will cross Panama , the Caribbean , Southern Africa and Madagascar be fore hurtling into space . Foreign affairs spokesman Pieter Swanepoel said neither had anything's department received any request or contacted the American authorities to find out what was happening with Cassini . ...
Extracted facts	• The Cassini space probe : due to be launched from Cape Canaveral in Florida of the United States at dawn • Local mass media quoted opponents of Cassini as saying at the weekend the mission will cross Panama • Foreign affairs spokesman Pieter Swanepoel said neither had anything's • department to find out what was happening with Cassini • Pieter Swanepoel : Foreign affairs spokesman • ...

(a) Extracting facts from the target corpus.

Encyclopedia entry	Cassini-Huygens is a joint NASA/ESA unmanned space mission intended to study Saturn and its moons. The spacecraft consists of two main elements: the Cassini orbiter and the Huygens probe. The spacecraft was launched on October 15 , 1997 and entered Saturn's orbit on July 1 , 2004. October 15, is the first spacecraft to orbit Saturn and just the fourth spacecraft to visit Saturn.
Extracted facts	1. Cassini - Huygens a joint NASA/ESA unmanned space mission intended to study Saturn and its moons 2. The spacecraft consists of two main elements 3. the Cassini orbiter the Huygens probe 4. The spacecraft entered Saturn's orbit on July 1, 2004 5. October 15, is the first spacecraft to orbit Saturn just the fourth spacecraft to visit Saturn 6. October 15, to visit Saturn

(b) Extracting facts from the reference corpus.

Re-ranked facts	• Cassini will be carrying 12 separate packages of scientific instruments a probe [3] • Saturn's largest moon [1] • department to find out what was happening with Cassini [3] • the instruments on Cassini to provide pictures of Saturn Nearly seven meters' rings moons radar to pierce the orange [1] • The Cassini space probe : due to be launched from Cape Canaveral in Florida of the United States at dawn [3] • ...

(c) Re-ranked facts from the target corpus (with the id of the most similar reference fact in brackets).

Table 1: Fact extraction and re-ranking in action.

4. EVALUATION

4.1 Experimental Setting

We applied the described method to find answers to the "other" questions and evaluated it within the TREC Question Answering track at TREC 2004 [19]. For the QA task, a list of questions was given, divided into 65 groups, each organized around a certain topic; examples include *James Dean*, *cataract* and *Teapot Dome scandal*. For each topic, a number of factoid questions were given, and an additional "other" question which requires as a response a list of important information nuggets regarding the topic. The important nuggets were set in advance by the assessors, and were divided into "essential" facts and less important "acceptable" ones. Each participant in the track returned a list of information nuggets for each topic.

The response was scored using the F-measure of precision and recall with recall three times more important. The recall measures the fraction of the essential nuggets returned, and the precision penalizes nuggets not considered essential or acceptable, and very long nuggets. More precisely, let E be the number of essential nuggets identified by the assessors; ER and AR are the number of nuggets returned by the system and judged as essential and acceptable, respectively;

length denotes the total length (the number of non-white-space characters) in the returned nuggets. Then

$$R = ER/E$$
$$allowance = 100 \cdot (ER + AR)$$
$$P = \begin{cases} 1, \text{ if } length < allowance \\ 1 - [(length - allowance)/length], \text{otherwise} \end{cases}$$
$$F = (10 \cdot P \cdot R)/(9 \cdot P + R)$$

In essence, this F-measure gives a higher importance to recall than precision, and rewards responses with lengths which are less than a per-topic threshold.

4.2 Results

In order to evaluate the effect of ranking nuggets using an external reference corpus, we included two versions of answers to "other" questions in the our official TREC QA runs, the baseline and a re-ranked version:

- In the baseline version we extracted nuggets from the target collection, as described above, and used prior estimates for the importance of the facts (I_{pr}) to rank the nuggets. We submitted 20 or less nuggets per topic (20 was an arbitrary threshold).
- For the re-ranked version, we used posterior estimates of the nugget importance (I_{post}) instead; also 20 or less facts were submitted per topic.

The results of the runs are given in Table 2. For comparison, the best system at TREC 2004 achieved an F-measure of 0.46, while the median F-measure over all 63 submitted runs is 0.184.

Measure	**Baseline**	**Re-ranked**
Precision	0.176	0.220 (+25%)
Recall	0.208	0.237 (+14%)
F-measure	0.184	0.210 (+14%)

Table 2: Evaluation results for "other" questions.

Note that the version of the F-measure used at TREC 2004 is biased towards recall. As is clear from Table 2, our re-ranking method substantially improves both recall and precision, but more so for precision.

A further per question breakdown of the change of performance in terms of F-measure is given in Figure 1 (top), indicating that while the gain in F-measure averaged over all questions is positive, there are questions whose score is affected negatively by our re-ranking mechanism. The results in Table 2 indicate that our re-ranking mechanism affects precision and recall differently; this is reflected in Figure 1 (middle) and (bottom), where we provide per-question breakdowns for precision and recall.

For 26 questions the F-measure of the original sentences (without re-ranking) is 0, preventing any improvement from our re-ranking method; in Figure 2 we present the breakdown for all but the 26 zero scoring questions.

4.3 A Closer Look

An analysis of the assessed runs revealed that often good nuggets were in the collection, but not in the top 20 documents we used for the extraction. Indeed, the threshold of

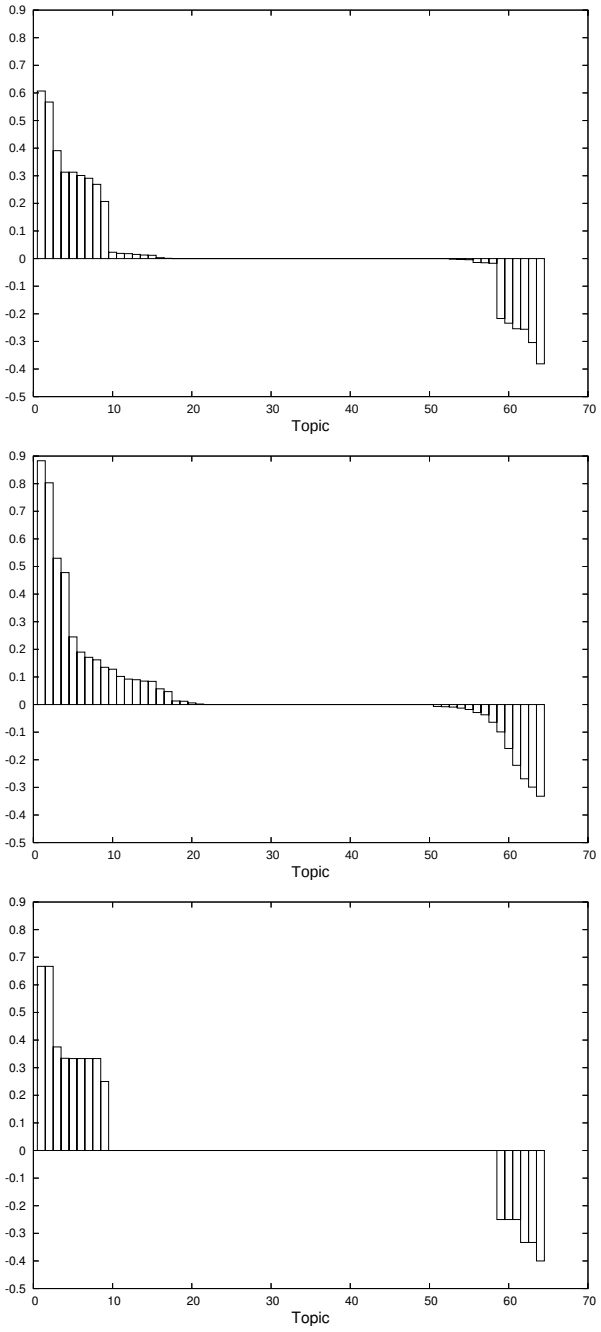

Figure 1: Per-question breakdown of effect of re-ranking, all questions: F-measure (top), precision (middle), and recall (bottom).

20 was set mainly for computational reasons, and further experiments with higher thresholds has shown clear improvements. Since the evaluation of new runs has to be done manually, we have no numerical support for this claim.

Another major source of errors was the similarity measure (normalized word overlap) used in our submitted runs. Because of its sparsity, often the decision about a match was based on a single common word, as, e.g., for the third nugget in Table 1(c). Again, experimenting with more suit-

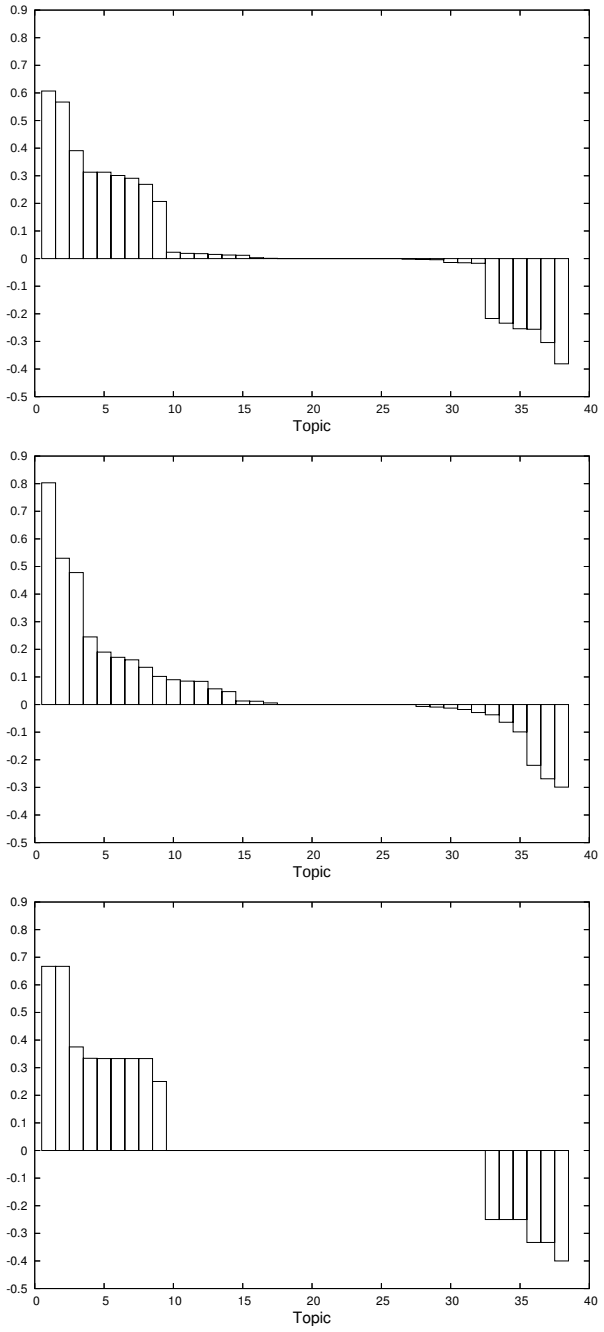

Figure 2: Per-question breakdown of effect of re-ranking, for questions with non-zero F-measure before re-ranking: F-measure (top), precision (middle), and recall (bottom).

able measures is hindered by the lack of automatic evaluation methodology: unlike the factoid questions at TREC, for "other" questions it is difficult to create patterns of correct answers. Developing such effective automatic evaluation methods is essential for improving the systems.

Re-ranking errors were also caused by our sentence splitting and anaphora resolution methods. For example, for the topic "*Carlos the Jackal*" one of the important nuggets "*the man known as Carlos the Jackal, once considered the world's most wanted terrorist, is serving a life sentence there*" was discarded after re-ranking, although the reference corpus provided the nugget "*on December 23 he was found guilty and sentenced to life imprisonment.*" Although both contain the key words *sentence* and *imprisonment*, the nugget from the target collection was too long for the similarity to be detected by our method. A better sentence splitter (capable of ignoring reduced relative clauses) could make nuggets shorter and the similarity more obvious. Another reason for discarding the snippet was the incorrectly resolved referential "*there.*" Had it been resolved to its true antecedent "*La Sante*," the snippet could have matched the reference nugget "*he was sent to La Santé de Paris prison to await trial.*"

4.4 Discussion

In our method, we assume availability of a reference corpus, a high-quality, clean and well-structured text collection, reflecting *the user's perspective* on which information is relevant or important for which topics. When faced with new information (in our case, coming from a different, unstructured, less reliable and less focused corpus), we use the reference collection to identify and rank new facts. Sentence similarity is used as a device to check how well a new bit of information matches the needs of users.

Our implementation of this model, as described in this paper, is far from complete. First of all, currently our system is capable of identifying facts from the target corpus which are very similar to those in the reference collection. But, users would probably also be interested in finding new facts, different from those the reference corpus can provide. To address this quite natural need, we can generalize the notion of fact similarity. Abstracting from concrete entities in the reference corpus, we can observe, for example, that if the user model considered the fact "*X was founded...*" important for the topic X, then the fact "*Y was founded...*" is likely to be important for the topic Y. Modifying the similarity metric to use information about, e.g., named entities and their types, we can use our reference corpus on a more abstract level and provide the user with both new and important information.

Second, the method we use to split long sentences (typical for newspaper text) into manageble facts is not very robust. It is based on full syntactic parsing and suffers from parsing errors, often producing ungrammatical and hardly interpretable nuggets. While parsing (identification of predicate-argument structure) can lead to a more informed estimation of fact similarity, more robust chunking methods should probably be used to present results to the user.

There are many other parts of the system that need attention: disambiguating entries in the reference corpus using previous questions of the user, improving the anaphora resolution module (see, e.g, nugget 5 in Table 1(b), where the pronoun *it* was incorrectly resolved to *October 15*) and extending it to handle definite NP anaphora.

5. CONCLUSIONS

We described a way to use high-quality semi-structured resources to model preferences of users for retrieval of short facts. By comparing facts extracted from a target collection to the information from a reference resource, we identify those facts that are potentially *important* for the user. Even with a simple word overlap-based similarity measure,

this method shows reasonable performance: applying it to answer "other" questions in the TREC 2004 QA track, we show substantial improvements over the baseline.

Our preliminary analysis of the TREC 2004 results suggest experimenting with more sophisticated sentence-level similarity measures and improving sentence splitting for extraction of atomic facts.

Acknowledgments. We thank David Ahn for his work on the anaphora resolver used in our experiments. This research was supported by the Netherlands Organization for Scientific Research (NWO) under project number 220-80-001. Maarten de Rijke was also supported by grants from NWO, under project numbers 365-20-005, 612.069.006, 612.-000.106, 612.000.207, 612.066.302, and 264-70-050.

6. REFERENCES

[1] D. Ahn, V. Jijkoun, J. Kamps, G. Mishne, K. Müller, M. de Rijke, and S. Schlobach. The University of Amsterdam at TREC 2004. In *TREC 2004 Conference Notebook*, Gaithersburg, Maryland USA, 2004.

[2] AltaVista Search Logs, URL: `ftp://ftp.archive.org/pub/AVLogs/`.

[3] E. Brill, J. Lin, M. Banko, S. Dumais, and A. Ng. Data-intensive question answering. In *Proc. 10th Text REtrieval Conference*, 2001.

[4] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336. ACM Press, 1998.

[5] J. Chu-Caroll, J. Prager, C. Welty, K. Czuba, and D. Ferrucci. A multi-strategy and multi-source approach to question answering. In *Proceedings TREC 2002*, 2003.

[6] C. A. Clarke, G. Cormack, G. Kemkes, M. Laszlo, T. Lynam, E. Terra, and P. Tilker. Statistical selection of exact answers (multitext experiments for trec 2002). In *Proceedings TREC 2002*, pages 823–831, 2003.

[7] M. De Boni and S. Manandhar. The use of Sentence Similarity as a Semantic Relevance Metric for Question Answering. In *Proceedings of the AAAI Symposium on New Directions in Question Answering*, 2003.

[8] P. Jaccard. The distribution of the flora of the alpine zone. *New Phytologist*, 11:37–50, 1912.

[9] P. Jourlin, S. Johnson, K. Spärck Jones, and P. Woodland. Improving retrieval on imperfect speech transcriptions. In *Proc. SIGIR '99*, pages 283–284, Berkeley, CA, 1999.

[10] J. Kamps. Improving retrieval effectiveness by reranking documents based on controlled vocabulary. In S. McDonald and J. Tait, editors, *Advances in Information Retrieval: 26th European Conference on IR Research (ECIR 2004)*, LNCS 2997, pages 283–295. Springer-Verlag, 2004.

[11] J. Kupiec. MURAX: a robust linguistic approach for question answering using an on-line encyclopedia. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 181–190. ACM Press, 1993.

[12] D. Lin. PRINCIPAR – an efficient, broad-coverage, principle-based parser. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING-94)*, 1994.

[13] D. Lin and P. Pantel. Discovery of inference rules for question answering. *Natural Language Engineering*, 7(4):343–360, 2001.

[14] J. Lin, A. Fernandes, B. Katz, G. Marton, and S. Tellex. Extracting Answers from the Web Using Knowledge Annotation and Knowledge Mining Techniques. In *Proc. 11th Text REtrieval Conference*, 2002.

[15] G. Mishne and M. de Rijke. Query formulation for answer projection. In *Proceedings ECIR 2005*, 2005.

[16] P. Ogilvie and J. Callan. Combining document representations for known-item search. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 143–150. ACM Press, 2003.

[17] I. Soboroff and D. Harman. Overview of the TREC 2003 novelty track. In *Proceedings TREC 2003*, pages 38–53, 2004.

[18] W. van Hage, M. de Rijke, and M. Marx. Information retrieval support for ontology construction and use. In S. McIlraith, D. Plexousakis, and F. van Harmelen, editors, *Proceedings 3rd International Semantic Web Conference (ISWC 2004)*, LNCS 3298, pages 518–533, 2004.

[19] E. Voorhees. Overview of the TREC 2003 question answering track. In *Proceedings Twelfth Text Retrieval Conference (TREC 2003)*, pages 54–68, 2003.

[20] E. Voorhees. Overview of the TREC-2004 question answering track. In *Proceedings 13th Text REtrieval Conference*, Gaithersburg, Maryland USA, To appear.

[21] J. Xu, A. Licuanan, and R. Weischedel. TREC 2003 QA at BBN: answering definitional questions. In *Proceedings TREC 2003*, pages 98–106, 2004.