Complex Interactive Question Answering Enhanced with Wikipedia

Ian MacKinnon
imackinn@cs.uwaterloo.ca
University of Waterloo
Waterloo, ON

Olga Vechtomova
ovechtom@engmail.uwaterloo.ca
University of Waterloo
Waterloo, ON

1 Introduction

In ciQA, templates are used with several bracketed items we call "facets" which are the basis of information being sought. This information is returned in the form of nuggets. Due to the concepts being sought having multiple terms to describe them, it becomes difficult to determine which sentences in the AQUAINT-2 news articles contain the query terms being sought, as they may be represented in the parent document by a variety of different phrases still making reference to the query term. For example, if the term "John McCain" were being sought, it might appear in an article, however, the sentence which is the vital nugget may simply contain "Senator McCain"; an imperfect match.

Traditional query expansion[5] of facets would introduce new terms which are related but do not necessarily mean the same as the original facet. This does not always help the problem of query terms appearing in relevant documents but not relevant sentences within documents, it only introduces related terms which cannot be considered synonymous with the facet being retrieved. In this year's TREC, we hope to overcome some of this problem by looking for synonyms for facets using Wikipedia.

Many of the ciQA facets are proper nouns and most thesauri, such as WordNet, do not contain entries for these. Thus, a new manner of finding synonyms must be found. In recent years, several new approaches have been proposed to use Wikipedia as a source of lexical information[2, 7], as it can be downloaded in its entirety, and contains relatively high quality articles[3].

2 Wikipedia

Every article in Wikipedia represents a concept, and all links from other articles must have an anchor text over the link. We also know that there are Wikipedia guidelines for what the anchor text should be for a link, and that we can assume that, provided editors are following the rules, the anchor text of the link will be of high quality. As we can see from this excerpt from the Wikipedia manual of style[1]:

> "It is possible to link words that are not exactly the same as the linked article title, for example, [[English language—English]]. However, make sure that it is still clear what the link refers to without having to follow the link." -Wikipedia Manual of Style

The anchor texts which point to the article will contain other terms for the same concept, which are necessary to get a better understanding of what concepts are represented in the text.

In order to make use of this information, we built a parser to extract every link in the Wikipedia collection; some 45 million links. Using this information, we can see what article links to which, but more importantly, what text is used to link to an article, and with what frequency. As we can see in Table 1, there are several different articles with 'radio waves' as anchor text on links to it.

Similarly, once we have an article, we can examine what anchor text is used to label links to that article

[1] http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style_\%28links\%29

Table 1: Frequency of Links to Articles that have "radio waves" as anchor text

Article	Text Frequency
radio waves	72
radio frequency	10
Electromagnetic radiation	3
radio	2
Radio Waves (album)	1

Table 2: Frequency of Anchor Text for "Radio Waves" article

Anchor Text	Linking Frequency
radio waves	72
radio	4
radio wavelengths	2
airwaves	1
electromagnetic vibration	1
radio signals	1
radiofrequency electromagnetic radiation	1

by doing a reverse lookup on our listing, as we can see in Table 2. It is using this information that we can effectively find synonyms for the concept of "Radio Waves". We are also given a type of confidence measure for those synonyms in the form of the linking frequencies.

3 Wikipedia Article Selection for Facets

We have devised a method of using the anchor text within Wikipedia links in order to resolve a small set of concepts which are represented in a candidate sentence.

We define the algorithm to turn a facet into a list of concepts as follows:

1. Set window length to n.

2. For each possible position of window, check all anchor text in Wikipedia to see if the phrase or term is recognized. If it is, record the matching string and drop the words covered in the window from future consideration. See Fig 1.

3. Decrease the length of the window by one ($n = n-1$). If the window length is 1, do not look up stopwords in term dictionary, simply ignore. Go to step 2 if window length is greater than 0.

4. For terms extracted from the query, look at the frequency of that term when linking to different articles. If an article has a majority of the links with that term as anchor text pointing to it, resolve that article to be the most relevant article for that multi-word unit. If no article has more than half the links with that anchor text pointing to it, drop the multi-word unit from consideration, as the term is ambiguous. However, if the frequency of anchor text linking to that article is less than 2, it is ignored.

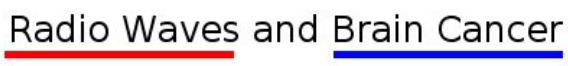

Figure 1: When a window recognizes a multi-word unit from the nugget, it saves it and drops the text from future consideration.

5. If there are multiple articles resolved for the query, select whichever article has the highest number of incoming links from all other Wikipedia articles to be the most relevant Wikipedia article for the given facet.See Fig 2.

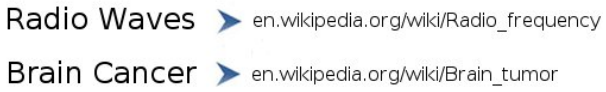

Figure 2: Multi-Word Units are resolved to whichever article has the most links with that anchor text.

In our experiments, we initially set $n = 5$.

4 Submitted Runs

4.1 UWinitBASE

The UWinitBASE run is based on the ciQA run UWATCIQA1 from 2006[8]. Only minor tuning parameter changes were made.

This base system selects sentences for a given query as follows:

1. Parse out the initial topic to get the 2 or 3 facets from the test topic.

2. Split apart each of facets into single terms, join all the terms together into one list, and perform a BM25[6] retrieval[2]. The top 50 documents from the AQUAINT newswire corpus are returned by the system and kept in order according to their BM25 scores.

3. Since we are interested in nuggets which contain information about the relationship between all of the facets, we need to ensure that all of the concepts noted in the test topic are represented in the document somewhere. For each of the returned documents, we wish to determine the validity of the document by ensuring at least one non-stopword from each of the facets exists in the document. Invalid documents are moved to the end of the list of 50 documents, effectively giving us an ordered list of valid documents sorted by BM25 followed by invalid documents sorted by BM25.

4. At this point we have a rather nice list of documents which likely contain information about the test topic, however, our goal is not to retrieve a list of relevant documents like many other TREC tasks, but rather to return a list of nuggets. To do this we break up the documents from the top 50 into their constituent sentences. In order to preserve the rankings provided to us by BM25, we keep the sentences in order in which their parent document occurred in the top 50 ranking.

5. We next require a method of ranking sentences for return. To do this, we will apply a score of 0,1,2, or 3 to a sentence depending on the number of facets which are represented in a candidate sentence. Each topic will have 2 or 3 facets containing a number of terms within them. For each facet, let us consider $\Gamma = \gamma_1 ... \gamma_n$ to be the set of non-stopword terms for a facet in a ciQA topic.

   A score is assigned to a candidate sentence S, by iterating through all the γ_i in Γ, and determining if any of the non-stopword stems of the terms exist in the sentence. If at least one exists, a nugget is said to be represented in the facet. More formally:

$$score(S, \Gamma) = \begin{cases} 1 & \text{if a } \gamma_i \in \Gamma \text{ exists in S} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

   We get the total score for S by taking the sum of the $score(S, \Gamma)$ for each facet in the topic.

   A sort is then performed on the list of sentences, but it is of great importance that the sort preserve the original ordering of the sentences with the same score. This allows for sentences which come from a document with a higher BM25 score to be ranked higher, given that they are likely more relevant to the test topic. In the likely event of ties, given that there are potentially hundreds of candidate sentences and only 3 possible scores in the highest case, ties are broken by applying a Brill tagger[1] to the text of the facets, and re-sorting the sentences based on the number of proper nouns from the facet which the sentence contains. The motivation of this being that proper nouns would be included in sentences that are more likely to be relevant, rather than just segue sentences in a news article.

6. The top n ranked sentences for each topic are output by the system. $n = 30$ seems to be a standard number of nuggets to return so as not to be too verbose but to allow for enough sentences to be returned that most of the vital nuggets would be accounted for in the sentences returned by the system.

[2] Using default parameters k=1.2, b=0.75

4.2 UWinitWIKI

This run utilizes the Wikipedia article parsing algorithm described earlier. Our intent is to incorporate the information from a facet's set of anchor text, A, in addition to the set of terms in the facet, Γ. A higher score is given to a candidate sentence, S, if it contains an anchor text term from A in it as opposed to simply a term from the facet. More formally:

$$score(S, \gamma_i) = \begin{cases} 1.2 & \text{if an } \alpha_i \in A \text{ exists in S} \\ 1 & \text{if a } \gamma_i \in \Gamma \text{ exists in S} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

The score of 1.2 is rather arbitrary. It just needed to be higher than 1, but low enough such that 2 matches from A would not be ranked higher than 3 from Γ.

From here, sentences are sorted according to score as before. The only difference from the baseline system is the integration of the A terms from the anchor text. The remaining issue is what method is used to select the articles from Wikipedia for the given facet, for which UWinitWIKI uses the automatic method described earlier. In the case where no article is available, the original metod from UWinitBASE is used, and the facets are broken up into their constituent terms.

4.3 UWfinalWIKI

This run was similar to UWbaseWIKI in that anchor text selected from Wikipedia articles chosen for each facet. However, articles were used not from the parsing algorithm but rather articles selected by NIST assessors. The form given to the NIST assessors can be seen in Fig 3. NIST assessors were given a "short list" of articles related to the facet, and were asked to select the minimum set which best described the facet.

The short list was given in order to make the job of the assessor easier to perform and not requiring an open ended search of the entire Wikipedia collection for relevant articles. The short lists consisted of up to 10 candidate articles obtained by taking articles from a Yahoo! search for the facet text restricted to the en.wikipedia.org domain, title lookups of all combinations of terms in the facets, and expanding disambiguation pages.

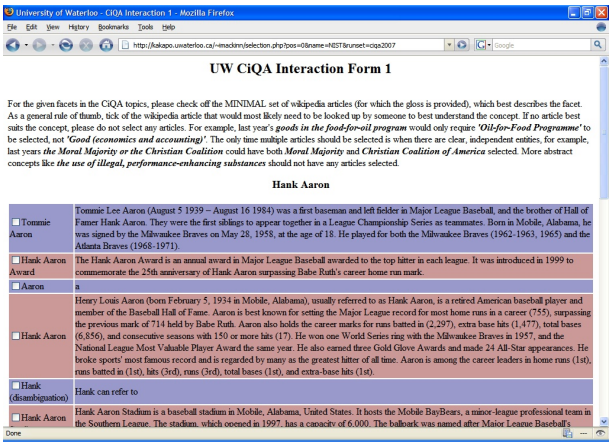

Figure 3: Screenshot of the user interaction to pick a articles from a short list.

Unfortunately, on the final day of the interaction phase, when only a small number of topics had been assessed, network access to the .uwaterloo.ca domain was cutoff due to an external outage and no further topics could be interactive with by the NIST assessors, meaning most of the nuggets returned for topics in this run are identical to UWinitBASE.

4.4 UWfinalMAN

In light of this network error, a manual run was submitted using articles selected by the author using the same selection interface. No other manual interaction was performed. It was necessary to give some kind of indication about the impact of humans picking the articles used for anchor text expansion rather than the automatic algorithm.

5 Results

The results for the submitted runs can be found in Table 3. Using the Nuggeteer[4] system on ciQA2006 data, a 10% performance increase was found using the systems from UWinitBASE and UWinitWIKI,

Table 3: Nugget Pyramid Scores along with Median for Runs

Run	F-Score
UWbaseINIT	0.388
UWbaseWIKI	0.388
Baseline Median	0.359
UWfinalWIKI	0.380
UWfinalMAN	0.386
Final Median	0.361

respectively. Compared to the results from this year, which are unchanged between the two baseline runs.

Looking at individual topic performance we see that contrasting UWinitBASE to UWinitWIKI, 11 topics were improved, 8 unchanged and 12 were worse off. Comparing UWiniWIKI to UWfinalMAN, 8 topics were improved, 14 unchanged, and 10 worse.

6 Conclusions

While the scores from all the runs were higher than the medians and generally performed very well, we were disappointed that there was no net benefit to using the synonyms derived from the Wikipedia link structure. Similar tests using the 2006 data showed a modest increase in F-scores.

However, different nuggets were returned by the two systems mean that it could be possible to identify trends in topics that performed poorly using the Wikipedia systems, and account for them. It may also be necessary to assign a weight to the anchor texts based on their frequencies as links. This would allow us to grade the synonym and incorporate it into the score more accurately.

Interestingly, human interaction yielded little benefit to this. It is likely that the human-selected articles which differed from the automatically selected ones were on topics which were not affected by the synonyms derived from the Wikipedia articles.

References

[1] E. Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–565, 1995.

[2] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of The Twentieth International Joint Conference for Artificial Intelligence*, Hyderabad, India, 2007.

[3] J. Giles. Internet encyclopaedias go head to head. *Nature*, 438(7070):900–901, December 2005.

[4] G. Marton and A. Radul. Nuggeteer: Automatic nugget-based evaluation using descriptions and judgements. In *Proceedings of NAACL/HLT*, 2006.

[5] S. E. Robertson. On term selection for query expansion. *Journal of Documentation*, 46(4):359–364, 1990.

[6] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[7] M. Strube and S. P. Ponzetto. Wikirelate! computing semantic relatedness using wikipedia. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, pages 1419–1424, Boston, Mass., July 2006.

[8] Vechtomova and Karamuftuoglu. Identifying relationships between entities in text for complex interactive question answering task. In *TREC*, 2006.