A Comparative Study of Word Co-occurrence for Term Clustering in Language Model-based Sentence Retrieval

Saeedeh Momtazi
Spoken Language Systems
Saarland University
saeedeh.momtazi
@lsv.uni-saarland.de

Sanjeev Khudanpur
Center for Language
and Speech Processing
Johns Hopkins University
khudanpur@jhu.edu

Dietrich Klakow
Spoken Language Systems
Saarland University
dietrich.klakow
@lsv.uni-saarland.de

Abstract

Sentence retrieval is a very important part of question answering systems. Term clustering, in turn, is an effective approach for improving sentence retrieval performance: the more similar the terms in each cluster, the better the performance of the retrieval system. A key step in obtaining appropriate word clusters is accurate estimation of *pairwise* word similarities, based on their tendency to co-occur in similar contexts. In this paper, we compare four different methods for estimating word co-occurrence frequencies from two different corpora. The results show that different, commonly-used contexts for defining word co-occurrence differ significantly in retrieval performance. Using an appropriate co-occurrence criterion and corpus is shown to improve the mean average precision of sentence retrieval form 36.8% to 42.1%.

1 Corpus-Driven Clustering of Terms

Since the search in Question Answering (QA) is conducted over smaller segments of text than in document retrieval, the problems of data sparsity and exact matching become more critical. The idea of using class-based language model by applying term clustering, proposed by Momtazi and Klakow (2009), is found to be effective in overcoming these problems.

Term clustering has a very long history in natural language processing. The idea was introduced by Brown et al. (1992) and used in different applications, including speech recognition, named entity tagging, machine translation, query expansion, text categorization, and word sense disambiguation. In most of the studies in term clustering, one of several well-know *notions of co-occurrence*—appearing in the same document, in the same sentence or following the same word—has been used to estimate term similarity. However, to the best of our knowledge, none of them explored the relationship between different notions of co-occurrence and the effectiveness of their resulting clusters in an end task.

In this research, we present a comprehensive study of how different notions of co-occurrence impact retrieval performance. To this end, the Brown algorithm (Brown et al., 1992) is applied to pairwise word co-occurrence statistics based on different *definitions* of word co-occurrence. Then, the word clusters are used in a class-based language model for sentence retrieval. Additionally, impact of corpus size and domain on co-occurrence estimation is studied.

The paper is organized as follows. In Section 2, we give a brief description of class-based language model for sentence retrieval and the Brown word clustering algorithm. Section 3 presents different methods for estimating the word co-occurrence. In Section 4, experimental results are presented. Finally, Section 5 summarizes the paper.

2 Term Clustering Method and Application

In language model-based sentence retrieval, the probability $P(Q|S)$ of generating query Q conditioned on a candidate sentence S is first calculated. Thereafter sentences in the search collection are ranked in descending order of this probability. For word-based unigram, $P(Q|S)$ is estimated as

$$P(Q|S) = \prod_{i=1...M} P(q_i|S), \qquad (1)$$

where *M* is the number of query terms, q_i denotes the i^{th} query term in Q, and *S* is the sentence model.

Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL, pages 325–328,
Los Angeles, California, June 2010.

For class-based unigrams, $P(Q|S)$ is computed using only the *cluster labels* of the query terms as

$$P(Q|S) = \prod_{i=1...M} P(q_i|C_{q_i}, S)P(C_{q_i}|S), \quad (2)$$

where C_{q_i} is the cluster that contains q_i and $P(q_i|C_{q_i}, S)$ is the emission probability of the i^{th} query term given its cluster and the sentence. $P(C_{q_i}|S)$ is analogous to the sentence model $P(q_i|S)$ in (1), but is based on clusters instead of terms. To calculate $P(C_{q_i}|S)$, each cluster is considered an atomic entity, with Q and S interpreted as sequences of such entities.

In order to cluster lexical items, we use the algorithm proposed by Brown et al (1992), as implemented in the SRILM toolkit (Stolcke, 2002). The algorithm requires an input corpus statistics in the form $\langle w, w', f_{ww'}\rangle$, where $f_{ww'}$ is the number of times the word w' is seen in the context w. Both w and w' are assumed to come from a common vocabulary. Beginning with each vocabulary item in a separate cluster, a bottom-up approach is used to merge the pair of clusters that minimizes the loss in *Average Mutual Information* (AMI) between the word cluster $C_{w'}$ and its context cluster C_w. Different words seen in the same contexts are good candidates for merger, as are different contexts in which the same words are seen.

While originally proposed with bigram statistics, the algorithm is *agnostic* to the definition of co-occurrence. E.g. if $\langle w, w'\rangle$ are verb-object pairs, the algorithm clusters verbs based on their selectional preferences, if $f_{ww'}$ is the number of times w and w' appear in the same document, it will produce semantically (or topically) related word-clusters, etc.

Several notions of co-occurrence have been used in the literature to cluster words, as described next.

3 Notions of Word Co-occurrence

Co-occurrence in a Document

If two content words w and w' are seen in the same document, they are usually topically related. In this notion of co-occurrence, how near or far away from each other they are in the document is irrelevant, as is their order of appearance in the document. *Document-wise* co-occurrence has been successfully used in many NLP applications such as automatic thesaurus generation (Manning et al., 2008)

Statistics of document-wise co-occurrence may be collected in two different ways. In the first case, $f_{ww'} = f_{w'w}$ is simply the number of documents that contain both w and w'. This is usually the notion used in ad hoc retrieval. Alternatively, we may want to treat each *instance* of w' in a document that contains an instance of w to be a co-occurrence event. Therefore if w' appears three times in a document that contains two instances of w, the former method counts it as one co-occurrence, while the latter as six co-occurrences. We use the latter statistic, since we are concerned with retrieving sentence sized "documents," wherein a repeated word is more significant.

Co-occurrence in a Sentence

Since topic changes sometimes happen within a single document, and our end task is sentence retrieval, we also investigate the notion of word co-occurrence in a smaller segment of text such as a sentence. In contrast to the document-wise model, *sentence-wise* co-occurrence does not consider whole documents, and only concerns itself with the number of times that two words occur in the same sentence.

Co-occurrence in a Window of Text

The *window-wise* co-occurrence statistic is an even narrower notion of context, considering only terms in a window surrounding w'. Specifically, a window of a fixed size is moved along the text, and $f_{ww'}$ is set as the number of times both w and w' appear in the window. Since the window size is a free parameter, different sizes may be applied. In our experiments we use two window sizes, 2 and 5, that have been studied in related research (Church and Hanks, 1990).

Co-occurrence in a Syntactic Relationship

Another notion of word similarity derives from having the same syntactic relationship with the context w. This *syntax-wise* co-occurrence statistic is similar to the sentence-wise co-occurrence, in that co-occurrence is defined at the sentence level. However, in contrast to the sentence-wise model, w and w' are said to co-occur only if there is a syntactic relation between them in that sentence. E.g., this type of co-occurrence can help cluster nouns that are used as objects of same verb, such as 'tea', 'water', and 'cola,' which all are used with the verb 'drink'.

To gather such statistics, all sentences in the corpus must be syntactically parsed. We found that a dependency parser is an appropriate tool for our goal: it

directly captures dependencies between words without the mediation of any virtual (nonterminal) nodes. Having all sentences in the parsed format, $f_{ww'}$ is defined as the number of times that the words w and w' have a parent-child relationship of *any syntactic type* in the dependency parse tree. For our experiments we use MINIPAR (Lin, 1998) to parse the whole corpus due to its robustness and speed.

4 Sentence Retrieval Experiments

4.1 Derivatives of the TREC QA Data Sets

The set of questions from the TREC 2006 QA track[1] was used as the test data to evaluate our models, while the TREC 2005 set was used for development.

The TREC 2006 QA task contains 75 question-series, each on one topic, for a total of 403 factoid questions which is used as queries for sentence retrieval. For sentence-level relevance judgments, the *Question Answer Sentence Pair* corpus of Kaisser and Lowe (2008) was used. All the documents that contain relevant sentences are from the NIST AQUAINT1 corpus.

QA systems typically employ sentence retrieval after initial, high quality document retrieval. To simulate this, we created a separate *search collection* for each question using all sentences from all documents relevant to the topic (question-series) from which the question was derived. On average, there are 17 relevant documents per topic, many *not* relevant to the question itself: they may be relevant to another question. So the sentence search collection is realistic, even if somewhat optimistic.

4.2 Corpora for Term Clustering

We investigated two different corpora[2], *AQUAINT1* and *Google n-grams*, to obtain word co-occurrence statistics for term clustering. Based on this we can also evaluate the impact of corpus size and corpus domain on the result of term clustering.

AQUAINT1 consists of English newswire text extracted from the Xinhua, the New York Times and the Associated Press Worldstream News Services.

The Google *n*-gram counts were generated from publicly accessible English web pages. Since there is no possibility of extracting document-wise, sentence-wise or syntax-wise co-occurrence statistics from the Google *n*-gram corpus, we only collect window-wise statistics to the extent available in the corpus.

Corpus	Co-occurrence	# Word Pairs
AQUAINT1	document	368,109,133
AQUAINT1	sentence	104,084,473
AQUAINT1	syntax	12,343,947
AQUAINT1	window-5	46,307,650
AQUAINT1	window-2	14,093,661
Google *n*-grams	window-5	12,005,479
Google *n*-grams	window-2	328,431,792

Table 1: Statistics for different notions of co-occurrence.

Table 1 shows the number of word pairs extracted from the two corpora with different definitions of co-occurrence. The statistics only include word pairs for which both constituent words are present in the 35,000 word vocabulary of our search collection.

4.3 Sentence Retrieval Results and Discussion

Sentence retrieval performance for term clustering using different definitions of word co-occurrence is shown in Figure 1. Since the Brown algorithm requires specifying the number of clusters, tests were conducted for 50, 100, 200, 500, and 1000 clusters of the term vocabulary. The baseline system is the word-based sentence retrieval model of Equation (1).

Figure 1(a) shows the *Mean Average Precision* (MAP) for class-based sentence retrieval of Equation (2) using clusters based on different co-occurrence statistics from AQUAINT1. Note that

(i) the best result achieved by sentence-wise co-occurence is better the best result of document-wise, perhaps due to more local and relevant information that it captures;

(ii) all the results achieved by syntax-wise co-occurrence are better than sentence-wise, indicating that merely co-occurring in a sentence is not very indicative of word similarity, while relations extracted from syntactic structure improve system performance significantly;

(iii) window-2 significantly outperforms all other notions of co-occurrence; i.e., the bigram statistics achieve the best clustering results. In comparison, window-5 has the worst results, with performance very close to baseline.

Although window-5 co-occurrence has been reported

[1] See `http://trec.nist.gov`.

[2] See catalog numbers LDC2002T31 and LDC2006T13 respectively at `http://www.ldc.upenn.edu/Catalog`.

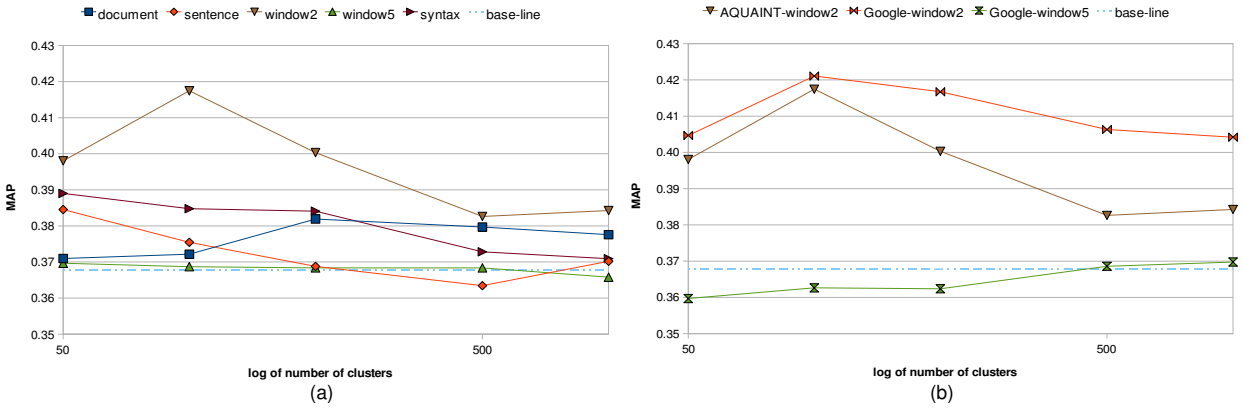

Figure 1: MAP of sentence retrieval for different word co-occurrence statistics from AQUAINT1 and Google *n*-grams.

to be effective in other applications, it is not helpful in sentence retrieval.

Figure 1(b) shows the MAP for class-based sentence retrieval of Equation (2) when window-wise co-occurrence statistics from the Google *n*-grams are used. For better visualization, we repeated the MAP results using AQUAINT1 window-2 co-occurrence statistics from Figure 1(a) in 1(b). Note that

(iv) window-2 co-occurrence statistics significantly outperform window-5 for the Google *n*-grams, consistent with results from AQUAINT1;

(v) Google *n*-gram window-2 co-occurrence statistics consistently result in better MAP than AQUAINT window-2.

The last result indicates that even though the Google *n*-grams are from a different (and much broader) domain than the test data, they significantly improve the system performance due to sheer size. Finally

(vi) Google *n*-gram window-2 MAP curve is flatter than AQUAINT window-2; i.e., performance is not very sensitive to the number of clusters.

The best overall result is from Google window-2 co-occurrence statistics with 100 clusters, achieving 42.1% MAP while the best result derived from AQUAINT1 is 41.7% MAP for window-2 co-occurrence with 100 clusters, and the MAP of the word-based model (baseline) is 36.8%.

5 Concluding Remarks

We compared different notions of word co-occurrence for clustering terms, using document-wise, sentence-wise, window-wise, and syntax-wise co-occurrence statistics derived from AQUAINT1. We found that different notions of co-occurrence significantly change the behavior of a sentence retrieval system, in which window-wise model with size 2 achieves the best result. In addition, Google *n*-grams were used for window-wise model to study the impact of corpus size and domain on the clustering result. The result showed that although the domain of the Google *n*-grams is dissimilar to the test set, it outperforms models derived from AQUAINT1 due to sheer size.

Acknowledgments

Saeedeh Momtazi is funded by the German research foundation DFG through the International Research Training Group (IRTG 715).

References

P.F. Brown, V.J.D. Pietra, P.V. Souza, J.C. Lai, and R.L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

K.W. Church and P. Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

M. Kaisser and J.B. Lowe. 2008. Creating a research collection of question answer sentence pairs with Amazon's mechanical turk. In *Proc. of LREC*.

D. Lin. 1998. Dependency-based evaluation of MINIPAR. In *Proc. of the Evaluation of Parsing Systems Workshop*.

C.D. Manning, P. Raghavan, and H. Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

S. Momtazi and D. Klakow. 2009. A word clustering approach for language model-based sentence retrieval in question answering systems. In *Proc. of ACM CIKM*.

A. Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proc. of ICSLP*.