Distributional Semantics for Answer Re-ranking in Question Answering[⋆]

Piero Molino, Pierpaolo Basile, Annalina Caputo,
Pasquale Lops, and Giovanni Semeraro

Dept. of Computer Science - University of Bari Aldo Moro
Via Orabona, 4 - I-70125, Bari (ITALY)
{piero.molino, pierpaolo.basile, annalina.caputo, pasquale.lops,
giovanni.semeraro}@uniba.it

Abstract. This paper investigates the role of Distributional Semantic Models (DSMs) into a Question Answering (QA) system. Our purpose is to exploit DSMs for answer re-ranking in QuestionCube, a framework for building QA systems. DSMs model words as points in a geometric space, also known as *semantic space*. Words are similar if they are close in that space. Our idea is that DSMs approaches can help to compute relatedness between users' questions and candidate answers by exploiting paradigmatic relations between words, thus providing better answer re-ranking. Results of the evaluation, carried out on the CLEF2010 QA dataset, prove the effectiveness of the proposed approach.

1 Introduction

Distributional Semantics Models (DSMs) represent word meanings through linguistic contexts. The meaning of a word can be inferred by the linguistic contexts in which the word occurs. The philosophical insight of distributional models can be ascribed to Wittgenstein's quote *"the meaning of a word is its use in the language"*. The idea behind DSMs can be summarized as follows: if two words share the same linguistic contexts they are somehow similar in the meaning. For example, analyzing the sentences "drink wine" and "drink beer", we can assume that the words "wine" and "beer" have similar meaning. Using that assumption, the meaning of a word can be expressed by the geometrical representation in a *semantic space*. In this space a word is represented by a vector whose dimensions correspond to linguistic contexts surrounding the word. The word vector is built analyzing (e.g. counting) the contexts in which the term occurs across a corpus. Some definitions of contexts may be the set of co-occurring words in a document, in a sentence or in a window of surrounding terms.

[⋆] This paper summarizes the main results already published in Molino, P., Basile, P., Caputo, A., Lops, P., Semeraro, G.: Exploiting Distributional Semantic Models in Question Answering. In: Sixth IEEE International Conference on Semantic Computing, ICSC 2012, Palermo, Italy, September 19-21, 2012. IEEE Computer Society 2012, ISBN 978-1-4673-4433-3.

This paper aims at exploiting DSMs for performing a task to which they have never been applied before, i.e. candidate answers re-ranking in Question Answering (QA), exploring how to integrate them inside a pre-existent QA system. Our insight is based on the ability of these spaces to capture paradigmatic relations between words which should result in a list of candidate answers related to the user's question.

In order to test the effectiveness of the DSMs for QA, we rely on a pre-existent QA framework called QuestionCube[1] [2]. QuestionCube is a general framework for building QA systems which exploits NLP algorithms, for both English and Italian, in order to analyze questions and documents with the purpose of allowing candidate answers obtained from the retrieved documents to be re-ranked by a pipeline of scorers. Scores assign a score to a candidate answer taking into account several linguistic and semantic features. Our strategy for exploiting DSMs consists in adding a new scorer to this pipeline, based on vector spaces built using DSMs. In particular, we propose four types of spaces: a classical Term-Term co-occurrence Matrix (TTM) used as baseline, Latent Semantic Analysis (LSA) applied to TTM, Random Indexing (RI) approach to reduce TTM dimension, and finally an approach which combines LSA and RI. The scorer will assign a score based on the similarity between the question and the candidate answers inside the DSMs.

2 Methodology

QuestionCube is a multilingual QA framework built using NLP and IR techniques. Question analysis is carried out by a full-featured NLP pipeline. The passage search step is carried out by Lucene, a standard off-the-shelf retrieval framework that allows TF-IDF and BM25 weighting. The question re-ranking component is designed as a pipeline of different scoring criteria. We derive a global re-ranking function combining the scores with CombSum. More details on the framework and a description of the main scorers is reported in [2]. The only scorers employed in the evaluation are: **Terms Scorer**, **Exact Sequence Scorer** and **Density Scorer**, a scorer that assign a score to a passage based on the distance of the question terms inside it. All the scorers have an enhanced version which adopts the combination of lemmas and PoS tags as features.

Our DSMs are constructed over a co-occurrence matrix. The linguistic context taken into account is a window w of co-occurring terms. Given a reference corpus[2] and its vocabulary V, a $n \times n$ co-occurrence matrix is defined as the matrix $\mathbf{M} = (m_{ij})$ whose coefficients $m_{ij} \in \mathbb{R}$ are the number of co-occurrences of the words t_i and t_j within a predetermined distance w. The $term \times term$ matrix $\mathbf{M}$, based on simple word co-occurrences, represents the simplest semantic space, called Term-Term co-occurrence Matrix (TTM). In literature, several methods to approximate the original matrix by rank reduction have been proposed. The aim of these methods varies from discovering high-order relations between entries to

[1] www.questioncube.com

[2] In our case the collection of documents indexed by the QA system.

improving efficiency by reducing its noise and dimensionality. We exploit three methods for building our semantic spaces: Latent Semantic Analysis (*LSA*), Random Indexing [1] (*RI*) and LSA over RI (*LSARI*). *LSARI* applies the SVD factorization to the reduced approximation of $\mathbf{M}$ obtained through RI. All these methods produce a new matrix $\hat{\mathbf{M}}$, which is a $n \times k$ approximation of the co-occurrence matrix $\mathbf{M}$ with n row vectors corresponding to vocabulary terms, while k is the number of reduced dimensions. We integrate the DSMs into the framework creating a new scorer, the **Distributional Scorer**, that represents both question and passage by applying addition operator to the vector representation of terms they are composed of. Furthermore, it is possible to compute the similarity between question and passage exploiting the cosine similarity between vectors using the different matrices.

3 Evaluation

The goal of the evaluation is twofold: (1) proving the effectiveness of DSMs into our question answering system and (2) providing a comparison between the several DSMs.

The evaluation has been performed on the *ResPubliQA 2010 Dataset* adopted in the *2010 CLEF QA Competition* [3]. The dataset contains about 10,700 documents of the European Union legislation and European Parliament transcriptions, aligned in several languages including English and Italian, with 200 questions. The adopted metric is the accuracy $a@n$ (also called $success@n$), calculated considering only the first n answers. If the correct answer occurs in the top n retrieved answers, the question is marked as correctly answered. In particular, we take into account several values of n =1, 5, 10 and 30. Moreover, we adopt the Mean Reciprocal Rank (MRR) as well, that considers the rank of the correct answer. The framework setup used for the evaluation adopts Lucene as document searcher, and uses a NLP Pipeline made of a stemmer, a lemmatizer, a PoS tagger and a named entity recognizer. The different DSMs and the classic TTM have been used as scorers alone, which means no other scorers are adopted in the scorers pipeline, and combined with the standard scorer pipeline consisting of the Simple Terms (ST), the Enhanced Terms (ET), the Enhanced Density (ED) and the Exact Sequence (E) scores. Moreover, we choosed empirically the parameters for the DSMs: the window w of terms considered for computing the co-occurrence matrix is 4, while the number of reduced dimensions considered in LSA, RI and LSARI is equal to 1,000.

The performance of the standard pipeline, without the distributional scorer, is shown as a baseline. The experiments have been carried out both for English and Italian. Results are shown in Table 1, witch reports the accuracy $a@n$ computed considering a different number of answers, the MRR and the significance of the results with respect to both the baseline (†) and the distributional model based on TTM (‡). The significance is computed using the non-parametric Randomization test. The best results are reported in bold.

Table 1. Evaluation Results for both English and Italian

		English					Italian				
	Run	a@1	a@5	a@10	a@30	MRR	a@1	a@5	a@10	a@30	MRR
alone	TTM	0.060	0.145	0.215	0.345	0.107	0.060	0.140	0.175	0.280	0.097
	RI	0.180	0.370	0.425	0.535	0.267‡	0.175	0.305	0.385	0.465	0.241‡
	LSA	**0.205**	**0.415**	**0.490**	0.600	**0.300**‡	0.155	0.315	0.390	0.480	0.229‡
	LSARI	0.190	0.405	**0.490**	**0.620**	0.295‡	**0.180**	**0.335**	**0.400**	**0.500**	**0.254**‡
combined	*baseline*	*0.445*	*0.635*	*0.690*	*0.780*	*0.549*	*0.445*	*0.635*	*0.690*	*0.780*	*0.549*
	TTM	0.535	0.715	0.775	0.810	0.614	0.405	0.565	0.645	0.740	0.539†
	RI	0.550	0.730	0.785	**0.870**	**0.637**†‡	0.465	**0.645**	**0.720**	**0.785**	0.555†
	LSA	**0.560**	0.725	**0.790**	0.855	**0.637**†	0.470	**0.645**	0.690	**0.785**	0.551†
	LSARI	0.555	**0.730**	**0.790**	**0.870**	0.634†	**0.480**	0.635	0.690	**0.785**	**0.557**†‡

Considering each distributional scorer on its own, the results prove that all the proposed DSMs are better than the TTM, and the improvement is always significant. The best improvement for the MRR in English is obtained by LSA (+180%), while in Italian by LSARI (+161%). Taking into account the distributional scorers combined with the standard scorer pipeline, the results prove that all the combinations are able to overcome the baseline. For English we obtain an improvement in MRR of about 16% with respect to the baseline and the result obtained by the TTM is significant. For Italian, we achieve a even higher improvement in MRR of 26% with respect to the baseline using LSARI. The slight difference in performance between LSA and LSARI proves that LSA applied to the matrix obtained by RI produces the same result of LSA applied to TTM, but requiring less computation time, as the matrix obtained by RI contains less dimensions than the TTM matrix.

Finally, the improvement obtained considering each distributional scorers on its own shows a higher improvement than their combination with the standard scorer pipeline. This suggests that a more complex method to combine scorers should be used in order to strengthen the contribution of each of them. To this purpose, we plan to investigate some learning to rank approaches as future work.

References

1. Kanerva, P.: Sparse Distributed Memory. MIT Press (1988)
2. Molino, P., Basile, P.: QuestionCube: a Framework for Question Answering. In: Amati, G., Carpineto, C., Semeraro, G. (eds.) IIR. CEUR Workshop Proceedings, vol. 835, pp. 167–178. CEUR-WS.org (2012)
3. Penas, A., Forner, P., Rodrigo, A., Sutcliffe, R.F.E., Forascu, C., Mota, C.: Overview of ResPubliQA 2010: Question Answering Evaluation over European Legislation. In: Braschler, M., Harman, D., Pianta, E. (eds.) Working notes of ResPubliQA 2010 Lab at CLEF 2010 (2010)