Exploring the Use of Word Embeddings and Random Walks on Wikipedia for the CogAlex Shared Task

Josu Goikoetxea, Eneko Agirre, Aitor Soroa
IXA NLP Group, University of the Basque Country, Basque Country
jgoicoechea009@ikasle.ehu.es, e.agirre@ehu.es, a.soroa@ehu.es

Abstract

In our participation on the task we wanted to test three different kinds of relatedness algorithms: one based on embeddings induced from corpora, another based on random walks on WordNet and a last one based on random walks based on Wikipedia. All three of them perform similarly in noun relatedness datasets like WordSim353, close to the highest reported values. Although the task definition gave examples of nouns, the train and test data were based on the Edinburgh Association Thesaurus, and around 50% of the target words were not nouns. The corpus-based algorithm performed much better than the other methods in the training dataset, and was thus submitted for the test.

1 Introduction

Measuring semantic similarity and relatedness between terms is an important problem in lexical semantics (Budanitsky and Hirst, 2006). It has applications in many natural language processing tasks, such as Textual Entailment, Word Sense Disambiguation or Information Extraction, and other related areas like Information Retrieval. Most of the proposed techniques are evaluated over manually curated word similarity datasets like WordSim353 (Finkelstein et al., 2002), in which the weights returned by the systems for word pairs are compared with human ratings.

The techniques used to solve this problem can be roughly classified into two main categories: those relying on pre-existing knowledge resources (thesauri, semantic networks, taxonomies or encyclopedias) (Alvarez and Lim, 2007; Yang and Powers, 2005; Hughes and Ramage, 2007; Agirre et al., 2009; Agirre et al., 2010) and those inducing distributional properties of words from corpora (Sahami and Heilman, 2006; Chen et al., 2006; Bollegala et al., 2007; Agirre et al., 2009; Mikolov et al., 2013).

Our main objective when participating in the CogAlex shared task was to check how a sample of each kind of technique would cope with the task. We thus selected one of the best corpus-based models to date and another approach based on random walks over WordNet and Wikipedia.

2 Word Embeddings

Neural Networks have become quite a useful tool in NLP on the last years, specially in semantics. A lot of models have been developed, but all of them share two characteristics: they learn meaning from non-labeled corpora and represent meaning in a distributional way. These models learn the meaning of words from corpora, and they represent it distributionally by the so-called embeddings. This embeddings are low-dimensional and dense vectors composed by integers, where the dimensions are latent semantic features of words.

We have used the Mikolov model (Mikolov et al., 2013) for this task, due to its effectiveness in similarity experiments (Baroni et al., 2014). This neural network reduces the computational complexity of previous architectures by deleting the hidden layer, and also, it's able to train with larger corpora (more than 10^9 words) and extract embeddings with larger dimensionality.

Zock/Rapp/Huang (eds.): Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon, pages 31–34,
Dublin, Ireland, August 23, 2014.

The Mikolov model has two variants: Continuous Bag of Words (CBOW) and Skip-gram. The first one is quite similar to the feedforward Neural Net Language Model, but instead of a hidden layer it has a projection layer; so, all the words are projected in the same position. Word order has thus no influence in the projection. Training criterion is as follows: knowing past and future words, it will predict the one in the middle.

The Skip-gram model is related to the previous one. The main difference is that it uses each current word as an input to a log-linear classifier with a continuous projection layer, and predicts words within a certain range before and after the current word.

In order to participate in this shared task, we have used the $word2vec$ tool[1]. On the one hand, we have used embeddings trained with the Skip-gram model on part of Google News corpus (about 100 billion words). The vectors have 300 dimensions and are publicly available[2]. On the other hand, we have adapted the $distance$ program in $word2vec$, so that its input is the test-set file of the shared task. The way $distance$ works is as follows:

- Reads all the vectors from the embeddings file, and stores them in memory.
- Reads the test-set file, and line by line
  - Saves the five entry words if they exist in vocabulary.
  - Dimension by dimension, sums five entries' embeddings into one vector, and normalizes it.
  - Calculates the semantic distance from the normalized vector to all words in vocabulary, and selects the closest ones.
  - Writes in output file the closest words along with their distances, writing the closest word first.

3 Random Walks on Wikipedia

In the last year there have been many attempts to apply graph based techniques to many NLP problems, including word sense disambiguation (Agirre et al., 2014) or measuring semantic similarity and relatedness between terms (Agirre et al., 2009). Those techniques consider a given Knowledge Base (KB) as a graph, where vertices represent KB concepts and relations among concepts are represented by edges.

For this particular task we represented WikiPedia as a graph, where articles are the vertices and links between articles are the edges. Contrary to other work using Wikipedia links (Gabrilovich and Markovitch, 2007; Milne and Witten, 2008), the use of the whole graph allows to apply algorithms that take into account the whole structure of Wikipedia. We applied PageRank and Personalized PageRank on the Wikipedia graph using freely available software (Agirre and Soroa, 2009; Agirre et al., 2014)[3].

The PageRank algorithm (Brin and Page, 1998) ranks the vertices in a graph according to their relative structural importance. The main idea of PageRank is that whenever a link from v_i to v_j exists in a graph, a vote from node i to node j is produced, and hence the rank of node j increases. Besides, the strength of the vote from i to j also depends on the rank of node i: the more important node i is, the more strength its votes will have. Alternatively, PageRank can also be viewed as the result of a random walk process, where the final rank of node i represents the probability of a random walk over the graph ending on node i, at a sufficiently large time. *Personalized PageRank* (Haveliwala, 2002) is a variant of the PageRank algorithm which biases the computation to prefer certain nodes of the graph.

Our method also needs a dictionary, an association between strings and Wikipedia articles. We construct the dictionary using article titles, redirections, disambiguation pages, and anchor text extracted from a Wikipedia dump[4]. Mentions are lowercased and all text between parenthesis is removed. If the mention links to a disambiguation page, it is associated with all possible articles the disambiguation page points to. Each association between a string and article is scored with the prior probability, estimated as the number of times that the mention occurs in the anchor text of an article divided by the total number of occurrences of the mention.

[1] `http://word2vec.googlecode.com/svn/trunk/`

[2] `https:\/\/docs.google.com/uc?id=0B7XkCwpI5KDYNlNUTTlSS21pQmM\&export=download`

[3] `http://ixa2.si.ehu.es/ukb`

[4] we used a 2013 Wikipedia dump to build the dictionary

The method to compute the answer for a given set of words is very simple. We just compute the Personalized PageRank algorithm over Wikipedia, initializing the walk using the set of given words, obtaining a probability distribution over all Wikipedia articles. We then choose the article with maximum probability, and return the title of the article as the expected answer.

Regarding PageRank implementation details, we chose a damping value of 0.85 and finish the calculation after 30 iterations. Some preliminary experiments on a related Word Sense Disambiguation task indicated that the algorithm was quite robust to these values, and we did not optimize them.

4 Development results

After running the random walks algorithms on the development data, it was clear that WordNet and Wikipedia were not sufficient resources for the task, and they were performing poorly. The embeddings, on the other hand, were doing a good job (accuracy of 14.1%, having returned a word on 1907 of the 2000 train instances). This is in contradiction with the results obtained in word relatedness datasets: for instance, in the WordSim353 dataset (Gabrilovich and Markovitch, 2007) we obtain Spearman correlations of 68.5 using random walks on WordNet, 72.8 using random walks on Wikipedia, and 71.0 using the embeddings.

One important difference between datasets like WordSim353 and the CogCalex data, is that in WordSim353, all words are nouns in singular. From a small sample of the CogaLex training data, on the contrary, we saw that only around 50% of the target words[5] are nouns, with many occurrences of grammatical words, and words in plural. Wikipedia only contains nouns, and even if WordNet contains verbs and adjectives, the semantic relations that we use are not able to check whether a meaning should be lexicalized as an adjective (absent in the dataset) or noun (absence). Note also that the random walk algorithm does not use co-occurrence data, and as such it is not able to capture that absent and minded are closely related as in "absent minded".

These differences between the WordSim 353 and the CogaLex data would explain the different behaviour of the algorithms. We would also like to mention that the definition of the task mentioned examples which are closer to the capabilities of WordNet and Wikipedia (e.g. given a set of words like "*gin, drink, scotch, bottle and soda*" the expected answer would be *whisky*). From the definition of the task, it looked as if the task was about recovering a word given a definition (as in dictionaries), but the actual data was based on the Edinburgh Association Thesaurus, which is a different kind of resource.

5 Test results

Given the development much better results of the embeddings, we submitted a run based on those. We obtained 16.35% accuracy, ranking fourth in the evaluation of all twelve submissions.

6 Conclusions

We tested three different kinds of relatedness algorithms: one based on embeddings induced from corpora, another based on random walks on WordNet and a last one based on random walks based on Wikipedia. All three of them perform similarly in noun relatedness datasets like WordSim353. Although the task definition gave examples of content nouns alone, the train and test data were based on the Edinburgh Association Thesaurus, and only around 50% of the target words were nouns. The embedding performed much better than the other methods in this dataset.

Acknowledgements

This material is based in part upon work supported by MINECO, in the scope of the CHIST-ERA READERS (PCIN-2013-002-C02-01) and SKATER (TIN2012-38584-C06-02) projects.

[5]The words that need to be predicted.

References

E. Agirre and A. Soroa. 2009. Personalizing PageRank for Word Sense Disambiguation. In *Proceedings of 14th Conference of the European Chapter of the Association for Computational Linguistics*, Athens, Greece.

E. Agirre, A. Soroa, E. Alfonseca, K. Hall, J. Kravalova, and M. Pasca. 2009. A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. In *Proceedings of annual meeting of the North American Chapter of the Association of Computational Linguistics (NAAC)*, Boulder, USA, June.

E. Agirre, M. Cuadros, G. Rigau, and A. Soroa. 2010. Exploring Knowledge Bases for Similarity. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May. European Language Resources Association (ELRA).

Eneko Agirre, Oier Lopez de Lacalle, and Aitor Soroa. 2014. Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1):57–88.

M.A. Alvarez and S.J. Lim. 2007. A Graph Modeling of Semantic Similarity between Words. *Proceedings of the Conference on Semantic Computing*, pages 355–362.

Marco Baroni, Georgiana Dinu, and Germn Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of ACL*.

D. Bollegala, Matsuo Y., and M. Ishizuka. 2007. Measuring Semantic Similarity between Words using Web Search Engines. In *Proceedings of WWW'2007*.

S. Brin and L. Page. 1998. The Anatomy of a Large-scale Hypertextual Web Search Engine. In *Proceedings of the seventh international conference on World Wide Web 7*, WWW7, pages 107–117, Amsterdam, The Netherlands, The Netherlands. Elsevier Science Publishers B. V.

A. Budanitsky and G. Hirst. 2006. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Computational Linguistics*, 32(1):13–47.

H. Chen, M. Lin, and Y. Wei. 2006. Novel Association Measures using Web Search with Double Checking. In *Proceedings of COCLING/ACL 2006*.

L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. 2002. Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, 20(1):116–131.

E Gabrilovich and S Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of IJCAI 2007*, pages 1606–1611, Hyderabad, India.

T.H. Haveliwala. 2002. Topic-sensitive PageRank. In *Proceedings of the 11th international conference on World Wide Web (WWW'02)*, pages 517–526, New York, NY, USA.

T. Hughes and D. Ramage. 2007. Lexical Semantic Relatedness with Random Graph Walks. In *Proceedings of EMNLP-CoNLL-2007*, pages 581–589.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of NAACL-HLT*, pages 746–751.

D. Milne and I.H. Witten. 2008. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *In Proceedings of the first AAAI Workshop on Wikipedia and Artifical Intellegence (WIKIAI'08)*, Chicago, I.L.

M. Sahami and T.D. Heilman. 2006. A Web-based Kernel Function for Measuring the Similarity of Short Text Snippets. *Proc. of WWW*, pages 377–386.

D. Yang and D.M.W. Powers. 2005. Measuring Semantic Similarity in the Taxonomy of WordNet. *Proceedings of the Australasian conference on Computer Science*.