Distinguishing Antonyms from Synonyms in Vector Space Models of Semantics

Bruna Thalenberg *

University of São Paulo
bruna.thalenberg@usp.br

Abstract. This research will investigate means to uncover differences between synonym and antonym pairs in vector space models of semantics 19 15 13. One of the main criticisms of such models is that they would be incapable of distinguishing between related words, such as antonyms, and similar ones, namely synonyms and hypernyms. However, Scheible et al. 17 proposed a method that could establish this difference, with promising but improvable results. We will adapt their study to Brazilian Portuguese, verifying if the results will be similar, and try to improve its accuracy. We will use the CETENFolha corpus, made available through Linguateca by the NILC research group, and the vectors will be created through gensim 14, the Python distribution of word2vec, originally created by Mikolov et al. 8 9.

Keywords: Distributional Semantics; Vector Space Models; Antonymy; Brazilian Portuguese

1 Introduction

Saussure, in his *Course in General Linguistics* 16, established the difference between the meaning and the value of a linguistic sign. While the first is related to a concept and is the counterpart to an acoustic image, the second is related to the system. According to the Swiss linguist, the value of a sign is established by its relation to other signs, accounting for negativity: a sign is what the others are not. This relation can be thought paradigmatically, in terms of substitution, or sintagmatically, in terms of concatenation: the value of a sign is given by the positions it can occupy, in opposition to the ones it cannot.

Distributional Semantics is a research area based on the premise that words with similar meanings occur in similar contexts 4. This way, important aspects of word meaning are described as a function of the set of contexts in which a word occurs. Thus, it aims to determine the value of words from the sintagmatic and paradigmatic relations they establish with others.

* I would like to thank my advisor, Marcelo Ferreira, for his constant guidance, and also the anonymous reviewers for helpful criticism. This work is part of an undergraduate 12-month research project ("Iniciação Científica") developed in the Department of Linguistics at the University of São Paulo, Brazil.

More recently, computational resources have been used to improve and implement this theory; one of its unfoldings being vector space models 19 15. In this approach, co-occurence matrices are created from corpora, and from them, vectors are extracted for each word and can then be compared.

The concept of vector is taken from mathematics, specifically from linear algebra. In the case of Distributional Semantics, its coordinates are equivalent to the number of times the target word co-occurs with another word, taken from the lines of a co-occurence matrix. In these matrices, then, we have in each cell the number of times the target word, represented in the lines, co-occurs with the neighbor word, represented in the columns. It is important to note that the size of the neighborhood of a word is arbitrarily defined, according to the purpose of the study: adjacent words can be neighbors, as can words in the same document. As for the number of columns, it is possible for a matrix to include only the words in a small text, although it is more common to use the 10.000 to 50.000 most frequent words in a corpus 5.

With the data obtained from these matrices, two types of similarity can be established: first-degree co-occurence, or sintagmatic association, between words that frequently appear near each other, and second-degree occurence, or paradigmatic association, between words that occur in similar contexts. The first one is easily extracted from the matrix, through the observation of its cells — words that frequently appear near each other will have a high value on their crossing. In turn, the second one requires some more work: through the observation of the generated vectors, i.e. the lines of the matrix, and the comparison between them, we can establish paradigmatically similar words. The vector space reveals itself to be a rich resource for the study of the value of a linguistic sign 15.

After the vectors are created, the similarity between two given words can be measured by the proximity of their vectors. For that to be possible, they are normalized using their scalar product, so that words with higher frequency are not preferred. Then, we can work out similar words using the cosine similarity of their normalized vectors. The cosine similarity is a measure of angle, not magnitude, hence it is ideal for measuring the proximity between two unit vectors 19. In the case of a vector space, it can vary between 0, for orthogonal vectors, and 1, for vectors that are exactly the same.

There is, nonetheless, a problem with the notion of semantic proximity: it is impossible to distinguish between synonyms and antonyms of a word, between co-hyponyms, or even between hyponyms and their hypernyms, since they usually occur in the same contexts. Their resulting vectors are, as expected, extremely similar and, as a consequence, their similarity measure also is 15.

Many were the solutions offered to this problem 6 18 10; albeit most of them need fixed structures — called contextual patterns — such as "x and not y" or "from x to y, or external resources, such as thesauri, to work. Although they perform very well, their use is somehow limited. In the case of contextual patterns, their occurence is not guaranteed, hampering the performance of the method with less frequent adjectives, as Scheible et al. pointed out 17. For the methods

based in external resources, the problem is even bigger. In addition to thesauri becoming dated in a very short period of time and not containing all the words in a language, their use compromises the applicability of vector spaces: if one of their functions is to create thesauri and WordNets automatically, they cannot need one to work properly.

A group of German researchers, however, obtained relative success using exclusively a vector space model, in a pioneer project 17. Guided by the hypothesis that not all parts of speech are useful for distinguishing antonyms from synonyms, they created different co-occurence matrices, one for each category, resulting in multiple vector spaces. They verified that even though nouns cannot be used for this distinction, other categories can — verbs being the most useful one. The obtained result is promising, but its accuracy rate of 70,6%, obtained from a Decision Tree classifier based on a single feature (standard-cosine or cosine-difference values), is not optimal for such a crucial distinction. Moreover, the experiment was only carried out with a German corpus, never being replicated with other languages, although the authors believe that similar results will be encountered.

This research project presents itself in this scope, aiming to adapt the study of Scheible et al. 17 to a Brazilian Portuguese corpus, to verify that the results will be, indeed, similar, and to try to improve the accuracy rate of the model. Since the model used by Scheible et al. is, according to Baroni et al.'s denomination 1, a count-based model, while our chosen software, gensim, is a predictive model, we will also re-run the experiment doing the appropriate adaptations for fair comparison.

2 Aims

- Verify means to distinguish antonyms from synonyms in a Vector Space Model, adapting the Scheible et al. 17 proposal to Brazilian Portuguese and to a predictive model.
- Construct a Brazilian Portuguese vector space model from which one could extract similarity measures between words.
- Compare our results to those obtained by Scheible et al. 17 with German.
- Compare the obtained results to those obtained through a contextual pattern 6 search.
- Verify the possibility of merging these two methods (Vector Space and contextual pattern search) in one algorithm, with the intention of improving the accuracy obtained.

3 Methodology and implementation

We will use the CETENFolha corpus to generate the co-occurence matrices. It is a Brazilian Portuguese corpus, already parsed, lemmatized, POS-tagged and divided by sentences, composed of fragments from the 1994 archive of Folha de

São Paulo, a newspaper, made available by NILC. It includes 25.475.272 tokens and 343.620 types.

The algorithm which will be used to create the Vector Spaces is gensim, the Python distribution of word2vec, originally created by Mikolov et al. 8 9. We chose to use gensim for it is the only one between the state-of-the-art open-source algorithms written in Python, a language we are comfortable using. It is also "one of the most widely used tools for building word vectors [...]" and while there are more sophisticated approaches available, it "remains a popular choice due to their efficiency and simplicity" 7. Nonetheless, we also plan to run GloVe, by Pennington et al. 12 for comparison. Word2vec (and, therefore, gensim) offers two models for unsupervised learning of the vectors: CBOW – continuous bag of words – and skip-gram, both of which ignore word order. Each of the models has two different training methods, with or without negative sampling. We will use the most recommended one, the skip-gram with negative sampling, as explained by Goldberg and Levy (2014) 3.

These models offered by word2vec have an advantage over the simple count-method we described in the introduction: those vectors are very sparse, and the computational cost for them when dealing with a big corpus is too high. Instead of capturing co-occurrence counts directly, then, both the CBOW and the skip-gram models learn vectors by starting with random numbers and then "iteratively making a word's embeddings more like the embeddings of neighboring words, and less like the embeddings of words that don't occur nearby" 5, and, in such way, they constitute recurrent neural network language models.

A skip-gram model learns, in fact, two separate vectors for each word: the word embedding $\boldsymbol{w}$ and the context embedding $\boldsymbol{c}$, which are encoded in two matrices – W and C. It then computes the probability of a word w_k occurring in a given position in reference to the target word w_j by computing the dot product between the the context vector for w_k and the target vector for w_j, and then normalizing it into probabilities using a *softmax* function. In negative sampling, the denominator of the *softmax* function through non-neighbour noise words according to their weighted unigram probability. The W and C are randomly initialized and their values are shifted while moving through the corpus as to maximize the objetive, using error backpropagation. This process results in vectors that have a high dot-product between neighboring words and a low dot-product with noise words 5.

Another advantage of word2vec/gensim is that it doesn't require a stopword list or any other treatment for very frequent words: it already implements an algorithm of subsampling, which improves the accuracy of learned vectors for rare words and accelerates learning by discarding words whose frequency is above a chosen threshold. Baroni et al. 1 show that predictive models have an overall performance much better than count-based models, giving us another good reason to choose gensim.

For the antonym and synonym pairs, we will filter the 100 most frequent adjectives on the corpus. Manually, we will remove those which do not have a contrasting term, and we will categorize the pairs. Using paper-based thesauri

like the ones of Fernandes (2002) 2 and Nascentes (1981) 11 and internet-based ones, such as Antônimos.com, Sinônimos.com and Antônimos e Sinônimos.com, we will list antonyms and synonyms for the obtained adjectives.

After filtering the sentences that contain the target words, we will train the model to obtain the vectors. We will use windows of different sizes, not exceeding five tokens in each side of the target word and not crossing sentence borders, in order to verify which is the most adequate extension of window for the study; and we will generate different spaces for each part of speech category — i.e. one for verbs, one for nouns, etc. — beside the complete one. In parallel, we will apply the contextual pattern filter proposed by Lin et al. 6 on the complete vector space.

Finally, we will verify if these methods are enough to determine the distance between antonyms using a classifier, based on a minimum score for the cosine similarity, obtained through the experiment. The results will be compared to the ones of Scheible et al. 17, from whom we adapted the experiment, and of Lin et al. 6, who used contextual patterns.

4 Implications

Distributional Semantics is still an underexplored area in Brazil. Nonetheless, it has a high applicability potential to Natural Language Processing technologies and to Computational Linguistics. One of its main advantages compared to other approaches, such as semantic webs and feature-based models, is its relative intrinsic independence to previous language and world knowledge, since it is entirely based on corpora.

For Natural Language Processing, some of the Vector Space Model applications are the elaboration of plagiarism detection softwares, automatic essay grading and automatic WordNet and thesaurus creation and improvement. Furthermore, it can be used to improve information retrieval and automatic translation systems. Considering these applications, the indistinguishability of antonyms and synonyms can be devastating. Therefore, solving this problem is of utmost importance.

From the point of view of my academic training, the development of this project will be fundamental to the acquisition and cristallization of my knowledge of linguistics (lexical semantics), mathematical (linear algebra) and computational (vector space models) concepts, allowing an interdisciplinary experience essential to those who would like to pursue a career in the field of Computational Linguistics. Moreover, this research will provide the crucial experience of dealing with a large corpus, something typical of this area.

References

Baroni, M., Dinu, G., Kruszewski, G.: Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In: Proceedings of ACL 2014 (52nd Annual Meeting of the Association for Computational Linguistics). pp. 238–247. ACL, East Stroudsburg, PA (2014)

Fernandes, F.: Dicionário de sinônimos e antônimos da língua portuguesa. Globo, São Paulo (2002)

Goldberg, Y., Levy, O.: word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. CoRR abs/1402.3722 (2014), http://arxiv.org/abs/1402.3722

Harris, Z.S.: Distributional structure. Word 2-3(10), 146–162 (1954)

Jurafsky, D., Martin, J.H.: Speech and Language Processing. Prentice Hall, New Jersey, 3rd edn. (draft), http://web.stanford.edu/~jurafsky/slp3/

Lin, D., Zhao, S., Qin, L., Zhou., M.: Identifying synonyms among distributionally similar words. In: Proceedings of the IJCAI. pp. 1492–1493. IJCAI, Acapulco (2003)

Ling, W., Dyer, C., Black, A., Trancoso, I.: Two/too simple adaptations of word2vec for syntax problems. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics (2015)

Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)

Mohammad, S.M., Hirst, G., Dorr, B.J., Turney, P.D.: Computing lexical contrast. Computational Linguistics 39(3), 555–590 (2013)

Nascentes, A.: Dicionário de sinônimos. Nova Fronteira, Rio de Janeiro (1981)

Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 1532–1543 (2014), http://www.aclweb.org/anthology/D14-1162

Řehůřek, R.: Scalability of Semantic Analysis in Natural Language Processing. Ph.D. thesis, Masaryk University, Brno (2011), http://radimrehurek.com/phd_rehurek.pdf

Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. pp. 45–50. ELRA, Valletta, Malta (May 2010), http://is.muni.cz/publication/884893/en

Sahlgren, M.: The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces. Ph.D. thesis, Stockholm University, Stockholm (2006), http://soda.swedish-ict.se/437/1/TheWordSpaceModel.pdf

Saussure, F.d.: Curso de Linguística Geral. Cultrix, São Paulo, 27 edn. (2007)

Scheible, S., Schulte im Walde, S., Springorum, S.: Uncovering distributional differences between synonyms and antonyms in a word space model. In: International Joint Conference on Natural Language Processing. pp. 489–497. Asian Federation of Natural Language Processing, Nagoya (2013), https://aclweb.org/anthology/I/I13/I13-1056.pdf

Turney, P.D.: A uniform approach to analogies, synonyms, antonyms, and associations. In: Proceedings COLING. pp. 905–912. ACL, Manchester (2008)

Widdows, D.: Geometry and Meaning. CSLI, Stanford (2004)