The original publication is available at: Springer
DOI: 10.1007/978-3-642-28885-2_26

A Bootstrapping Algorithm for Learning the Polarity of Words

António Paulo Santos[1], Hugo Gonçalo Oliveira[2], Carlos Ramos[1], and Nuno C. Marques[3]

[1] GECAD, Institute of Engineering - Polytechnic of Porto, Portugal
[2] CISUC, University of Coimbra, Portugal
[3] DI-FCT, Universidade Nova de Lisboa, Monte da Caparica, Portugal
pgsa (at) isep.ipp.pt, hroliv (at) dei.uc.pt, csr (at) isep.ipp.pt, nmm (at) di.fct.unl.pt

Abstract. Polarity lexicons are lists of words (or meanings) where each entry is labelled as *positive*, *negative* or *neutral*. These lists are not available for different languages and specific domains. This work proposes and evaluates a new algorithm to classify words as *positive*, *negative* or *neutral*, relying on a small seed set of words, a common dictionary and a propagation algorithm. We evaluate the *positive* and *negative* polarity propagation of words, as well as the *neutral* polarity. The propagation is evaluated with different settings and lexical resources.

Keywords: polarity of words, polarity lexicon, sentiment analysis, lexicon expansion, opinion mining

1 Introduction

Sentiment analysis, also known as opinion mining, often relies on polarity lexicons, also known as sentiment lexicons (e.g. [13], [3]). These lexicons are lists of words (e.g. [10], [11], [12]) or lists of meanings (e.g. [1], [8]) where each entry is labelled with their **a priori polarity**. For instance, even without knowing the context, words like *love*, *peace*, *fun* and *success* are usually labelled with a *positive prior polarity*, while words like *hate*, *war*, *bore*, *failure* are labelled with a *negative prior polarity*, and words like *people*, *table*, and *tree* are labelled as *neutral* in lexicons that also consider the *neutral* polarity. In a subsequent step this a priori polarity can be then used to determine the **contextual polarity** [13].

Many works (e.g. [10], [11]) classify words as *positive* or *negative*. However besides *positive* and *negative* connotations, words might have no connotation at all, which means they have *neutral* polarity. In this work, we extend the algorithm originally presented in [14], by considering the *neutral* class and representing the dictionary as an undirected graph. The method relies on a small seed set of words, a common dictionary represented as a graph, and a propagation algorithm.

The paper is organized as follows. In section 2 we review the related work and in section 3 we present the proposed propagation algorithm. In section 4 we evaluate the algorithm and finally, we conclude and discuss on further work, in section 5.

2 Background

In [11], 9,107 words are extracted from a common online dictionary and represented as a *directed graph*. On that graph, nodes represent words and edges represent the semantic relation between words. With 5 *positive* and 5 *negative* seed words, with manually labelled polarity, the authors applied a simple propagation and voting algorithm. Starting from the seed nodes, the algorithm visits every word in the graph by a breadth-first traversal and their polarity is iteratively propagated to the unlabelled words.

The representation of a dictionary varies. Some authors have chosen, to represent it as a *directed graph* (e.g. [2], [7], [11]), while others as an *undirected graph* (e.g. [10]). Particularly in [11], the dictionary is represented as a *directed graph* in order to preserve the original structure of the dictionary. In this work, we considered the property of symmetry of some semantic relations (e.g. synonymy, antonymy) and therefore we used an undirected representation. This representation relies on the fact that on a binary relation R between two words, if *word1 is related to word2* then *word2 is related to word1*, at least in some contexts. For instance, if *word A is a synonym of B*, *B* is as well a *synonym of A*.

3 The Polarity Propagation Algorithm

In this section we present the polarity propagation algorithm for classifying words as *positive*, *negative* and *neutral*. Assuming that we have a dictionary represented as an undirected graph (Fig. 1, at left), in which, each number represents a word, we get the list of classified words (graph on the right), by applying the following steps:

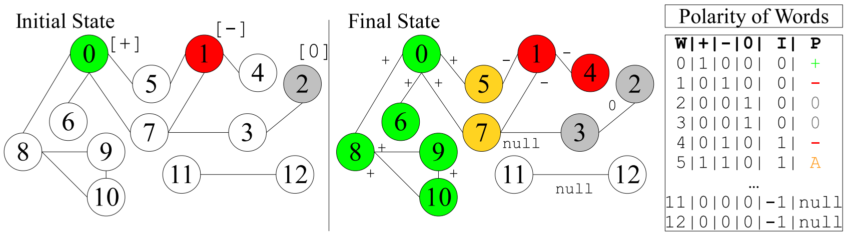

Fig. 1. Polarity propagation (*+, -, 0, A, null = positive, negative, neutral, ambiguous, and no polarity; w = word; I. = Iteration; P = final polarity*). Close to each node and edge is the polarity propagated to its neighbour.

1. Associate a *positive* (C_+), a *negative* (C_-), a *neutral* (C_0) and *iteration* (I) counter to each word. Initialize the first three counters to 0 and the fourth with a *negative* value. The *iteration* counter will record the shortest distance between each word and the closest seed word (the number of edges between them). This distance is useful to re-run the algorithm and apply a weighted propagation, as we will discuss in step 3.
2. Label a set of words $W = \{w_1, \ldots, w_n\}$ that should ideally contain the same number of *positive*, *negative* and *neutral* words. This is done by incrementing the respective counter with a *positive* value. Set the *iteration* counter of each word to 0 (this means that they are seed words). Finally, add all these words in a queue Q.
3. Retrieve the first word w_1 from the queue Q and get all their unvisited neighbours $Nb_{w1} = \{nb_1, \ldots, nb_m\}$. If $Nb_{w1} = \{\}$ go to step 4, else:
   (a) Propagate the *positive*, *negative* or *neutral* polarity of word w_1 to each one of its neighbours nb_i ($i = 1$ to m), by incrementing the counter of each neighbour, according to the following rules:

(1)	`If` $w_1 > 0 \ \wedge \ w_1 \longrightarrow nb_i$	`Then`	$pos. \Leftarrow pos. + 1 * Weight$
(2)	`If` $w_1 < 0 \ \wedge \ w_1 \longrightarrow nb_i$	`Then`	$neg. \Leftarrow neg. + 1 * Weight$
(3)	`If` $w_1 = 0 \ \wedge \ w_1 \longrightarrow nb_i$	`Then`	$neu. \Leftarrow neu. + 1 * Weight$
(4)	`If` $w_1 > 0 \ \wedge \ w_1 \dashrightarrow nb_i$	`Then`	$neg. \Leftarrow neg. + 1 * Weight$
(5)	`If` $w_1 < 0 \ \wedge \ w_1 \dashrightarrow nb_i$	`Then`	$pos. \Leftarrow pos. + 1 * Weight$
(6)	`If` $w_1 = 0 \ \wedge \ w_1 \dashrightarrow nb_i$	`Then`	$neu. \Leftarrow neu. + 1 * Weight$
(7)	`If` $w_1 = ambigous$	`Then`	*do nothing*

   Where:
   - Word w_1 is *positive* ($w_1 > 0$), *negative* ($w_1 < 0$), or *neutral* ($w_1 = 0$) if the respective counter is a unique maximum. Otherwise, the polarity of word w_1 is ambiguous and therefore the last rule is applied.
   - $w_1 \longrightarrow n_i$, represents any semantic relation between word w_1 and its neighbour n_i, where the polarity should be maintained (e.g. synonymy relation). $w_1 \dashrightarrow w_1$, represents any semantic relation that inverts the polarity (e.g. antonymy relation).
   - The $Weight$ variable should be set to 1 if we want to apply an unweighted propagation approach. If we want to apply a weighted propagation approach the $Weight$ should decrease as iteration increases, as discussed in [11].

   (b) For each neighbour nb_i of w_1 that does not exist on queue Q set the iteration counter to *iteration counter of* $w_1 + 1$ and add it to the end of Q. If n_i already exists on Q don't do nothing.
4. Mark word w_1 as visited.
5. If the Q is not empty, go to step 3, otherwise the algorithm ends (step 6).

6. At this step we have a list of words, each classified as:

`null`	`If` $C_+ = C_- = C_0 = 0$
`+`	`If` $C_+ > C_-$ `AND` $C_+ > C_0$
`-`	`If` $C_- > C_+$ `AND` $C_- > C_0$
`0`	`If` $C_0 > C_+$ `AND` $C_0 > C_-$
`Ambiguous`	`Otherwise`

In addition to the polarity, the values of the counters can be used to compute the polarity strength. For instance, the polarity strength of a positive word can be computed as the ratio of $C_+ \ / \ (C_+ + C_- + C_0)$.

4 Evaluation

4.1 Lexical Resources Used

In our experiments, we used a synonymy graph established by the semantic relations extracted from three dictionaries, namely *PAPEL 2.0*[4] [6], *Wikcionário.PT*[5] and *Dicionário Aberto*[6]. All the semantic relations were extracted in the scope of the Onto.PT project[7] [5] [4] and are available as triples in the form **lexical-item1 SEMANTIC-RELATION lexical-item2**, in which lexical-item is a lemmatized word or a multiword expression. In PAPEL 2.0 there are 79,161 synonymy triples containing 43,375 distinct lexical items, and, from Wikcionrio.PT and DA, there are 24,542 and 47,387 triples, containing 19,839 and 40,583 distinct lexical items respectively.

After building the graph and applying the propagation algorithm we evaluated the classified words using the following datasets:

– **SentiLex-PT01**[8] [12]: a publicly available sentiment lexicon for Portuguese. It contains 6,321 adjective lemmas classified in one of three classes: *positive* (+), *negative* (-), *neutral* (0). From those, 3,585 are manually classified and 2,736 are automatically classified. We used the entries manually classified.
– **Human 1**: a list of 460 Portuguese words (148 nouns, 107 verbs, 200 adjectives and 5 adverbs) manually classified by a native speaker. Each word was classified in one of three classes: *positive* (+), *negative* (-), and *neutral* (0).
– **Human 2**: the same previous dataset annotated by a different person.

The words in the datasets *Human 1* and *Human 2* were classified out of context and in a domain independent way. The annotator was instructed to classify each word according its primary thought. The goal was to try to capture the polarity of the words in most contexts, whenever possible (e.g. words such as *beautiful*, *peace* tend to be *positive* in most contexts, words such as *war* tend to be *negative*, and words such as *car*, *people* tend to be *neutral*).

The inter-annotator agreement for *Human 1* and *Human 2* datasets was 76.30% (Cohen's Kappa = 0.61). According to a commonly cited scale this value of the kappa is in the substantial agreement range (0.61-0.80) [14]. The obtained agreement can be further increased, for instance, by providing to the annotators words within sentences. However, for this study the obtained value is sufficient to measure the agreement of the algorithm in respect to those two annotators.

4.2 Results and Discussion

The goal of this experiment was to evaluate the classification performance of the propagation algorithm using: *PAPEL 2.0*, *Wikcionário.PT* and *Dicionário Aberto*. Experiment settings: *graph type* = undirected; *semantic relations* = synonymy; *num. of seed words* = 4(+), 4(-), 4(0); *evaluated classes* = +, -, 0.

Table 1 shows the results for 9 experiments. Each experiment was performed for 10 runs, varying the seed words. The seed words were retrieved from a list of about 150 seed words (50 (+), 50 (-), 50 (0)), most of them from the manually labelled entries in SentiLex-PT01.

[4] Available at http://www.linguateca.pt/PAPEL/
[5] http://pt.wiktionary.org/
[6] http://www.dicionario-aberto.net
[7] Available at http://ontopt.dei.uc.pt/index.php?sec=recursos
[8] Available at http://xldb.fc.ul.pt/wiki/SentiLex-PT01_in_English

Table 1. Average results for 9 experiments, 10 runs each, varying the seed words ($\overline{Class}$ = average number of classified words; $\overline{Eval}$ = average number of evaluated words; $\overline{Acc}$ ± SD = average accuracy ± standard deviation).

Eval.Dataset	PAPEL			D.Aberto			Wik.PT		
	$\overline{Class}$	$\overline{Eval}$	$\overline{Acc}$ $\pm SD$	$\overline{Class}$	$\overline{Eval}$	$\overline{Acc}$ $\pm SD$	$\overline{Class}$	$\overline{Eval}$	$\overline{Acc}$ $\pm SD$
SentiLex	30,044	2,338	**66 ±5.89**	20,391	1,188	**57 ±9.47**	11,069	1,034	**52 ±11.56**
Human 1	30,045	393	54 ±4.20	20,391	252	44 ±9.32	11,069	226	41 ±10.17
Human 2	30,045	393	60 ±9.24	20,391	252	46 ±21.30	11,069	226	41 ±18.03

According to the resource used, the best accuracy was obtained with *PAPEL*. These results seem to be related with the number of synonymy relations because the best results were obtained with the resource with more synonymy relations (79,161 relations) and the worst results with the resource with less synonymy relations (Wik.PT with 24,541 relations).

According to the evaluation dataset, the higher accuracy was obtained with *SentiLex*. Part of the reason may be because when we are using *SentiLex*, we are evaluating just adjectives. The accuracy of 66.03% is very close from the accuracy of 67% reported in [12] for the same task and evaluation dataset.

In conclusion we would like to draw attention to some aspects. First, since the inter-human agreement between *Human 1* and *Human 2* was 76.30%, the 0-76% scale seems more proper than the 0-100% scale, when analysing these results. Second, these are results considering words classified by the algorithm of all the iterations. Considering only words from the first iterations (e.g. until no further than 4th iteration) improve the results. Third, we just used 12 seed words. It is expected that increasing the number of words would improve the results.

5 Conclusion and Future Work

As any other approach that relies only on words to build or expand general purpose lexicons are limited. In order to reduce this limitation, we intend to adapt the current method to classify words in a domain or topic-specific way. Relying in word senses such as in *SentiWordNet* [1] for English, it may be also a possible future direction. As future work, we can also determine the contextual polarity in larger units of text like sentences, taking advantage of models similar to those used in part-of-speech tagging [9].

Acknowledgements

António Paulo Santos is supported by the FCT grant SFRH/BD/47551/2008.
Hugo Gonçalo Oliveira is supported by the FCT grant SFRH/BD/44955/2008 co-funded by FSE.
Nuno C. Marques wishes to thank InspirennovIT for sponsoring this work in the context of Best Supplier project[9].

References

1. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: P. of the 7th Conf. on Lang. Resources and Evaluation (LREC'10). vol. 25, pp. 2200–2204. Valletta, MT (2010)
2. Blair-Goldensohn, S., Hannan, K., Mcdonald, R., Neylon, T., Reis, G.A., Reynar, J.: Building a Sentiment Summarizer for Local Service Reviews. Electrical Engineering (2008)
3. Ding, X., Liu, B., Yu, P.: A holistic lexicon-based approach to opinion mining. In: Proceedings of the international conference on Web search and web data mining. pp. 231–240. ACM (2008)
4. Gonçalo Oliveira, H., Antón Pérez, L., Costa, H., Gomes, P.: Uma rede léxico-semântica de grandes dimensões para o português, extraída a partir de dicionários electrónicos. Linguamática 3(2), 23–38 (December 2011)
5. Gonçalo Oliveira, H., Gomes, P.: Onto.PT: Automatic Construction of a Lexical Ontology for Portuguese. In: In Proceedings of 5th European Starting AI Researcher Symposium (STAIRS 2010), IOS Press. pp. 199–211 (2010)
6. Gonçalo Oliveira, H., Santos, D., Gomes, P.: Extracção de relações semânticas entre palavras a partir de um dicionário: o PAPEL e a sua avaliação. Linguamática 2(1), 77–93 (2010)

[9] Portuguese National Strategic Reference Framework — *QREN/Programa Operacional de Factores de Competitividade*, proposal n. 18627 — 06/2010 SI I&DT supported by European Regional Development Fund.

7. Jijkoun, V., Hofmann, K.: Generating a Non-English Subjectivity Lexicon : Relations That Matter. Computational Linguistics pp. 398–405 (2009)
8. Maks, I., Vossen, P.: Different Approaches to Automatic Polarity Annotation at Synset Level. In: Proceedings of the First International Workshop on Lexical Resources, WoLeR 2011. pp. 62–69 (2011)
9. Marques, N., Lopes, G.: Tagging With Small Training Corpora. In: Proceedings of the International Conference on Intelligent Data Analysis (IDA01). pp. 63–72. Lecture Notes in Computer Science, Cascais, Portugal. Springer Verlag. (2001)
10. Rao, D., Ravichandran, D.: Semi-supervised polarity lexicon induction. Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics on EACL 09 pp. 675–682 (2009)
11. Santos, A.P., Ramos, C., Marques, N.C.: Determining the Polarity of Words through a Common Online Dictionary. In: 15th Portuguese C. on Artificial intelligence. LNCS, vol. 7026, pp. 649–663. Springer (2011)
12. Silva, M.J., Carvalho, P., Costa, C., Sarmento, L.: Automatic Expansion of a Social Judgment Lexicon for Sentiment Analysis. Technical Report. TR 10-08. University of Lisbon, Faculty of Sciences, LASIGE (2010)
13. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. Computational Linguistics pp. 1–41 (2011)
14. Viera, A.J., Garrett, J.M.: Understanding interobserver agreement: the kappa statistic. Family Medicine 37(5), 360–363 (2005)