View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by El Servicio de Difusión de la Creación Intelectual

Vector-based word representations for sentiment analysis: a comparative study

M. Paula Villegas[1], M. José Garciarena Ucelay[1], Juan Pablo Fernández[1], Miguel A. Álvarez-Carmona[2], Marcelo L. Errecalde[1], Leticia C. Cagnina[1,3]

[1]LIDIC, Universidad Nacional de San Luis, San Luis, Argentina
[2]Language Technologies Laboratory, INAOE, Puebla, México
[3]CONICET, Argentina
{villegasmariapaula74, mjgarciarenaucelay, miguelangel.alvarezcarmona, merrecalde, lcagnina}@gmail.com

Abstract. New applications of text categorization methods like opinion mining and sentiment analysis, author profiling and plagiarism detection requires more elaborated and effective document representation models than classical Information Retrieval approaches like the Bag of Words representation. In this context, word representation models in general and vector-based word representations in particular have gained increasing interest to overcome or alleviate some of the limitations that Bag of Words-based representations exhibit. In this article, we analyze the use of several vector-based word representations in a sentiment analysis task with movie reviews. Experimental results show the effectiveness of some vector-based word representations in comparison to standard Bag of Words representations. In particular, the Second Order Attributes representation seems to be very robust and effective because independently the classifier used with, the results are good.

Keywords: text mining, word-based representations, text categorization, movie reviews, sentiment analysis

1 Introduction

Selecting a good document representation model is a key aspect in text categorization tasks. The usual approach, named *Bag of Words* (BoW) model, considers that words are simple indexes in a term vocabulary. In this model, originated in the information retrieval field, documents are represented as vectors indexed by those words. Each component of a vector (document) represents the weight that the corresponding word has associated in that document. Well known limitations of the BoW representations are the sparseness of the resulting vectors and the loss of any information about the locations of words within documents.

Several proposals from the computational linguistic area have attempted mitigating the above mentioned limitations by considering more elaborated *word representations* [1]. In those approaches, words are considered as first-class objects that allow "more

semantic" comparisons among words and, in consequence, better categorization results.

Two word representations that have gained increasingly interest in the last years are the ones usually referred as *distributional* and *distributed* word representations [1]. Both approaches represent words as vectors that capture contextual information of the corresponding word within the documents. Nevertheless, they differ in the way those vectors are obtained, with an emphasis of distributed representations in *learning* word representations, typically using neural networks models. The latter approach is usually referred as *word embedding*.

On the other hand, sentiment analysis and opinion mining is an area receiving increasing attention from both, industry and academia. In this kind of problems, several of the above mentioned word representation methods have been applied with varying performance [2, 3]. However, to the best of our knowledge, most of those studies have focused on a particular word representation method, with limited comparison to other more elaborated word representations.

In the present work, this research gap is addressed by considering and comparing six effective word representations in a sentiment analysis task related to movie reviews. We focus on vector-based word representations by considering four methods of the distributional area (SOA, LSA, LDA, DOR) and one representative of the distributed representation approach (Word2Vec). Our study takes BoW as baseline and the result analysis is carried out on a subset of the IMDB Review Dataset.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the representations used in our comparative study. In Section 3, we describe the original sentiment analysis dataset and the proposed corpus used in our experiments. Section 4 shows the settings and the results corresponding to the experimental study. Finally, in Section 5 some conclusions are drawn and future works are proposed.

2 Vector-based word representations

In vector-based word representations (VWRs), each word has associated a *vector*. Each dimension's value corresponds to a *feature*, named *word feature*, which might have a semantic or grammatical interpretation [1]. VWRs are supposed to overcome limitations of the BoW model by allowing to capture richer relational structure of the lexicon [2]. This is achieved by encoding continuous (non-binary) similarities between words as distance or angle between word vectors in a high-dimensional space.

This section presents some basic explanations of the VWRs used in our work, four from the distributional representation area (subsections 2.2 to 2.6) and one representative of the distributed approach (subsection 2.7). Space limitations prevent us from giving detailed explanations of the methods and involved formulas but the interested reader can obtain them from the cited references. This section also includes in subsection 2.1 a short description of the standard BoW approach used as baseline in our study.

2.1 Bag of Words (BoW)

The traditional Bag of Words (BoW) representation is one of the most used in text categorization tasks. This popular representation is simple to implement, fast to obtain and can be used under different weighting schemes. However, the ordering of the words in the document is ignored, and the semantic and conceptual information are lost. As its name indicates, the document is represented as a bag of words and only the number of occurrences of each word is maintained. Formally, a document d is represented by the vector of weights $d_{bow} = \langle w_1, w_2, \ldots, w_n \rangle$ where w_i depends on the weighting scheme selected (*tf*, *tf-idf*, *Boolean*, etc.) and n is the size of the vocabulary of the dataset [4]. Often, the vectors of this representation could be very sparse if the documents are enough different.

2.2 Second Order Attributes (SOA)

SOA is a low dimensional representation with a high level of representativeness [5] that constructs a space of profiles from which vectors of terms are built. Then, using those vectors, the representation of the documents is obtained with respect to the same profiles space. Firstly, SOA identifies the profiles according to the categories to be used for the classification of the data. Then, for each term in the vocabulary, a value indicating the relationship (term frequencies in the documents belonging to the profile) between that term with each profile, is calculated and saved into a vector. Finally, the document representation is obtained adding all the vectors corresponding to all the terms included in the document, weighted by the relative frequency of that term in the document.

2.3 Latent Semantic Analysis (LSA)

LSA is a method for representing the contextual-usage meaning of words. LSA can associate words and its contribution to automatically generated concepts (topics) [6]. LSA assumes that words that are close in meaning will occur in similar pieces of text [7]. This is usually named the *latent space*, where documents and terms are projected to produce a reduced topic based representation. LSA is built from a matrix M where m_{ij} is typically represented by the *tf-idf* weight of the word i in document j. LSA uses the Singular Value Decomposition (SVD) and makes a reconstruction with K (topics) dimensions. It makes the best possible reconstruction of the M matrix with the less possible information and noise [8].

2.4 Latent Dirichlet Allocation (LDA)

LDA is a generative probabilistic model for collections of discrete data. LDA is a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of K topics. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities [9]. Basically, LDA observes each document, and randomly assigns each word in the document to one of the K topics. To improve the topics distribution LDA for each word in each document

assumes that this word is in a wrong topic and tries to fit the word in others topics maximizing the probability that the word is with others words with the same context. This process is repeated several times until the distribution of topical not change substantially.

2.5 Document Occurrence Representation (DOR)

DOR is an effective distributional representation based on the idea of how the semantic of a document can be described with a function of the bag of terms occurring on it, likewise the semantic of a term can be described with a function of the bag of the documents in which the term is [10]. Therefore, the more frequent the term is in a document, the more important that document is to characterize the semantic of that term. For each term in the vocabulary, DOR constructs vectors of weights considering the contribution of each document in the collection to the semantic of the term. Then, the representation of a document is obtained adding all the vectors of those terms occurring in the document [11] considering a weighting scheme.

2.6 Word2Vec

Word2Vec [12] is a representation learning method used to obtain distributed representations of words (also named *words embedding*s). The method learns a model using few labeled documents and many unlabeled documents with a neural network algorithm. The goal is to obtain word vectors with meaningful characteristics, that is, related words are in the same group if these appear in similar contexts, have similar meaning and/or have semantic relationships. The trained model is based on feature vectors (one for each word in the vocabulary) with a maximum number of dimensions. Then, for representing a document, the method searches for the words included in the text and uses the word vectors to construct the averaged vector that will be the Word2Vec representation.

3 Sentiment analysis

The sentiment analysis is a complex and challenging task in machine learning. When people write about their emotions, they can use sarcasm, some ideas can be ambiguous, and they use non common words. Then, the analysis of the sentiment of a text is not a trivial task. The sentiment label of a text can be categorical (positive or negative), continuous (a number indicating the level of positivity polarity, for example) or multidimensional (a combination of several types of labels). The first one was adopted in this work, so a document will have a positive or negative polarity.

Next, we describe the IMDB Review Dataset, a corpus containing movie reviews for sentiment analysis. Then, we present the subset of the original corpus used in our experiments.

3.1 Original corpus

IMDB Review Dataset [2] is a collection of 50000 movie reviews extracted from the popular movies, TV and celebrities content site www.imdb.com. The reviews are labeled as positive or negative regarding the sentiment polarity of the content. Both categories have the same number of reviews, that is, 25000 are positive and 25000 are negative reviews. In order to perform classification tasks, the dataset has been divided into train and test sets. Then, the training set has 25000 reviews labeled with the corresponding polarity and the testing set has also 25000 but unlabeled reviews [13].

3.2 Proposed corpus

We used only the labeled set of the IMDB Review Dataset because we need the category of each review for testing the classifiers with the different representations. We proposed a new corpus to use in this work which was constructed from the original IMDB Review Dataset training set. Our corpus is also divided in training and testing sets. Our train (named here as $IMDB_{train}$) and our test (named here as $IMDB_{test}$) sets were made selecting randomly the 90% of the reviews for the first one (constructing the model) and the 10% for the second one (testing), respectively. We maintain the same distribution of documents in each category (positive and negative) such as the original dataset, that is, the same number of reviews. Thus, $IMDB_{train}$ has 22500 reviews and $IMDB_{test}$ has 2500 reviews.

4 Experiments

4.1 Settings

We used the proposed corpus (Section 3.2) with 25000 movie reviews for all the experiments. We performed a pre-processing which included converting all the reviews to lowercase and removing stop words, numbers, punctuation marks and any special character. Then, we obtained the different representations with the following settings:

- BoW: the word vectors were weighted by the relative frequency of the word in each document. Only the 5000 most frequent terms were considered.

- Word2Vec: we kept the default parameters such as were suggested in [13], that is, the minimum word count is 40, word vector's dimensionality is 300, context window size is 10 and down sample setting for frequent words is 1e-3.

- LSA and LDA: we used k = 300 for the number of topics. After performing preliminary experiments, we obtained the best accuracy with this value.

– SOA: this representation has no parameters to set.

– DOR: this representation maintains in memory huge dimensional matrices and its computation (number of terms by number of documents) can be impossible to calculate even using modern computers. For this reason, we previously performed a study of information gain over the attributes (words) of $IMDB_{train}$. In that study, we found that adjectives were mainly the type of words which ranked first in the information gain analysis. This fact is illustrated in Table 1, which shows an example of the first 10 words with the highest information gain values. As we can observe from Table 1, almost all the words are adjectives. Hence, we only considered the adjectives as terms of each document for this representation. The weighting scheme used in these experiments was the Boolean as our first approach.

Table 1. The first 10 words with the highest information gain value obtained from $IMDB_{train}$.

	Term	Information Gain		Term	Information Gain
1	bad	0.063703	6	excellent	0.021465
2	worst	0.051894	7	terrible	0.020994
3	waste	0.033709	8	worse	0.019704
4	great	0.032522	9	wonderful	0.01893
5	awful	0.032153	10	stupid	0.017746

The experiments were performed using Naïve Bayes (NB) and LibLINEAR (LL) classifiers. We constructed the models with $IMDB_{train}$ and validated it using $IMDB_{test}$. We utilized the accuracy as measure to evaluate the performance of the classifications.

4.2 Results

The results of the experiments are shown in Table 2. The highest accuracy value is highlighted in bold. We decided to use BoW representation as a baseline to compare among the different representations.

Table 2. Accuracy obtained with Naïve Bayes (NB) and LibLinear (LL) classifiers for all the proposed document representations.

	Document Representations					
Classifiers	BoW	SOA	Word2Vec	LSA	LDA	DOR
NB	73.00	87.08	74.68	68.17	54.53	58.76
LL	83.76	86.96	87.12	**88.52**	70.73	78.60
Average	78.38	**87.02**	80.90	78.35	62.63	68.68

As it can be seen in Table 2, the best result was achieved by LSA representation with LibLINEAR classifier such as has previously demonstrated to perform well for personality recognition [14]. However, LSA with Naïve Bayes was over the baseline. In fact, for almost all representations, the results obtained using Naïve Bayes are below those obtained using LibLINEAR with exception of SOA.

The experiments with SOA representation shows high accuracy values for both classifiers and these ranks first if we consider the average of performance with both classifiers. Thus, we can say SOA seems to be the most robust representation with respect to the classification algorithms. Additionally, SOA has been shown to be a very robust and effective representation to classify other types of texts as well [15, 16].

On the other hand, Word2Vec, obtained good results with the LibLINEAR classifier using default parameters. This performance could be better if we change the parameters settings; due to the fact that the different model parameters can affect the quality of the Word2Vec representation.

Otherwise, LDA and DOR (last two columns in Table 2) did not perform well, both obtained accuracy values under the baseline with both classification algorithms. LDA obtained the lowest results indicating the unsuitability of this representation for this task. DOR combined with LibLINEAR is only a 5% below the baseline although this value is just above the average of the baseline. These poor results can be due to the version of DOR used in this work, because it is the most basic one, as we described in Section 4.1.

We can conclude with this comparative study that LDA, Word2Vec and SOA are the most adequate representations (over the six tested) for the sentiment analysis in movie reviews. In particular, LDA and Word2Vec representation seem to be quite dependent to the classification algorithms used with. Conversely, SOA seems to be very robust in addition to the efficiency in the performance. This can be observed in the results of Table 2 in which the values obtained with Naïve Bayes and LibLINEAR are quite similar.

5 Conclusions and future work

This article presents a comparative study about different word representations for the sentiment analysis on movie reviews. In particular, we were interested in those representations based on vectors of features (words in our case). We analyzed distributional representations such as BoW, SOA, LDA, LSA and DOR, and a distributed representation such as Word2Vec. We can conclude with this preliminary study that with some effective representations (LSA and Word2Vec) the algorithms obtain good results although these depend on the classifier used. On the other hand, with SOA representation we can obtain enough good results independently the classifier used.

For future work we plan to make a similar experimental study but using the complete IMDB Review Dataset, that is the 50000 movie reviews, in order to see if the conclusions about the good performance of the LSA, Word2Vec and SOA representations are still the same. We also propose to execute DOR using other types of parts of speech like nouns, verbs and articles, and other weighting schemes in the construction of the representation. For Word2Vec representation we plan to test with different parameters, not using just the default ones proposed by the authors. We are interested to study the stability and the execution time for all the representations.

6 References

1. Turian, J., Ratinov, L., Bengio, Y.: Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10). Association for Computational Linguistics, pp. 384--394. Stroudsburg, PA, USA (2010)

2. Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, N. Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT '11), Vol. 1. Association for Computational Linguistics, pp. 142--150. Stroudsburg, PA, USA (2011)

3. Mesnil, G., Mikolov, T., Ranzato, M., Bengio, Y.: Ensemble of Generative and Discriminative Techniques for Sentiment Analysis of Movie Reviews. arXiv preprint arXiv:1412.5335. (2014)

4. Feldman, R., Sanger, J.: The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, (2007)

5. López-Monroy, A. P., Montes-y-Gómez, M., Escalante, H. J., Villaseñor-Pineda, L., Villatoro-Tello, E.: INAOE's Participation at PAN'13: Author Profiling Task Notebook for PAN at CLEF 2013. In: CLEF (Working Notes), CEUR-WS.org, (2013)

6. Landauer, T. K., Dumais, S. T.: A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. Psychological Review, vol. 104(2), pp. 211--240. (1997)

7. Landauer, T., Foltz, P., Laham, D.: An introduction to latent semantic analysis. Discourse processes, vol. 25(2-3), pp. 259--284. (1998)

8. Landauer, T. K., McNamara, D. S., Dennis, S., Kintsch, W.: Handbook of latent semantic analysis. Psychology Press. (2013)

9. Blei, D. M., Ng, A. Y., Jordan, M. I.: Latent dirichlet allocation. In: Advances in neural information processing systems, pp. 601--608. (2001)

10. Lavelli, A., Sebastiani, F., Zanoli, R.: Distributional term representations: an experimental comparison. In: Proceedings of the thirteenth ACM international conference on Information and knowledge management. pp. 615--624. ACM. (2004)

11. Cabrera, J. M., Escalante, H. J., Montes-y-Gómez, M.: Distributional term representations for short-text categorization. In: International Conference on Intelligent Text Processing and Computational Linguistics. pp. 335--346. Springer Berlin Heidelberg. (2013)

12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representations in Vector Space, CoRR. abs/1301.3781. (2013)

13. Kaggle competition, https://www.kaggle.com/c/word2vec-nlp-tutorial/

14. Álvarez-Carmona, M. A., López-Monroy, A. P., Montes-y-Gómez, M., Villaseñor-Pineda, L., Escalante, H. J.: INAOE's Participation at PAN'15: Author Profiling task. In: CLEF (Working Notes), CEUR-WS.org. (2015)

15. Rangel, F., Rosso, P., Moshe Koppel, M., Stamatatos, E., Inches, G.: Overview of the author profiling task at pan 2013. In: CLEF Conference on Multilingual and Multimodal Information Access Evaluation. CELCT. 352-365. (2013)

16. Rangel, F., Rosso, P., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., Daeleman, W.: Overview of the 2nd a uthor profiling task at pan 2014. In: CEUR Workshop Proceedings, Vol. 1180. CEUR Workshop Proceedings. pp. 898--927. (2014)