Document embeddings for Arabic Sentiment Analysis

Amira Barhoumi[1,2] Yannick Estève[1] Chafik Aloulou[2] Lamia Hadrich Belguith[2]

(1) LIUM, Université du Maine, 72000 Le Mans
amira.barhoumi.etu@univ-lemans.fr , yannick.esteve@univ-lemans.fr
(2) MIRACL, Université de Sfax, Tunisie
amirabarhoumi29@gmail.com , chafik.aloulou@fsegs.rnu.tn ,l.belguith@fsegs.rnu.tn

Abstract. Research and industry are more and more focusing in finding automatically the polarity of an opinion regarding a specific subject or entity. Paragraph vector has been recently proposed to learn embeddings which are leveraged for English sentiment analysis. This paper focuses on Arabic sentiment analysis and investigates the use of paragraph vector within a machine learning techniques to determine the polarity of a given text. We tested some preprocessing method, and we show that light stemming enhance the performance of classification.

Keywords: Sentiment analysis, document embedding, paragraph vector, arabic language.

Introduction

With the widespread of Internet and the revolution of social networks, any person could follow his opinion and express his feelings and emotions regarding various topics, products, ideas, persons, etc. Many academic and industrial efforts are focusing on analyzing opinions and sentiments by investigating automatic techniques to extract convenient information.

Sentiment Analysis (SA) involves building systems that recognize the opinion expressed in a textual unit. It aims mainly to identify the subjectivity and polarity of a given text. Generally, the polarity consists of positive, negative or neutral, with or without their strength. SA and its applications have spread to many languages and most of the works deal with Indo-European ones. Indeed, several researches have been carried out for the English language. However, few works have been done for Arabic. In this work, we are interested in Arabic language.

In this paper, we present an application of sentiment analysis to the Arabic language. The main contributions of this work are as follows: (1) we measure the efficiency of distributed representation for Arabic sentiment analysis ASA, (2) we evaluate the performance of neuronal techniques for sentiment classification.

The rest of the paper is structured as follows. Section 2 discusses some related works dedicated mainly to Arabic. In section 3, we present our methodology for ASA. We report, in section 4, our experimental framework and discuss the obtained results. Finally, we conclude in section 5 and give some outlooks to future work.

Related work

Nowadays, sentiment analysis is becoming very interesting[1] due to the explosion of the number of internet users and the proliferation of social networks.

The largest amount of SA researches has been carried out for the English language. There are few works have been done for other languages. Recently, there has been a considerable effort to develop SA systems for the Arabic language. In this section, we focus on works dedicated for ASA. Most of the existing methods in sentiment analysis can be divided into three categories: knowledge based approach, machine learning based approach and hybrid approach.

The knowledge based approach uses lexicon or patterns. [4] proposed an approach based on a local grammar which contains patterns that extract sentiment from a given document. [3] followed the same approach based on patterns. For works based on lexicon, we quote the work of [5]. They manually construct a lexicon that contains 4815 words (1942 positive words and 2873 negatives ones). Their system compute the number of positive and negative words in a text in order to generate the overall polarity. Another work is that of [6] who implemented a tool which determine the subjectivity, the polarity and the strength of an opinion. They used two general lexicons and 16 specific lexicons (8 for positive polarity and 8 for negative polarity). For the strength computation, they manually added a score between 1 and 10 to each term in the lexicon. Another work has been done in [25] where the authors presented a lexicon based approach for MSA. First, a lexicon has been built by applying a semi automatic method. Then, the lexicon entries were used to detect opinion words and assign to each one a sentiment class. [26] built a sentiment lexicon of about 120,000 Arabic words and created a SA system on top of it. They reported a 86.89% of classification accuracy.

Machine learning based approach views SA as a classification task. Annotated data sets are used to train classifiers. [7] proposed a system that performs subjectivity and sentiment analysis for social media using morphological features. [8] compared Support Vector Machines SVM, Naive Bayes NB classifiers and neural networks which are trained on Opinion Corpus for Arabic OCA [18] and ACOM corpus [9] with different combinations. Another machine learning approach was used in [18] where they build the corpus OCA which consists of movie reviews written in Arabic. They also created an English version translated from Arabic and called EVOCA. SVM and NB classifiers are then used to create SA systems for both languages. For instance, SVM gives 90% F-measure on OCA compared to 86.9% on EVOCA.

In multi-way sentiment analysis, [10] performed a multi-class classification, using a scale from 1 to 5 to measure polarity. They tested SVM, decision tree C4.5, decision table J48, KNearest Neighbors KNN, NB, MultiNaive Bayes MNB and voting (a combination of KNN, decision tree and NB). They concluded that MNB is more efficient. The authors did a flat classification, i.e there is only one level in the hierarchy. However, [11] shows that a hierarchical classification of the multi-way sentiments is better than an ordinary flat classification. They have implemented two hierarchical structures: one with two levels and the other with four levels. They tested SVM, NB, KNN and decision tree techniques. They concluded that KNN is more efficient.

[1] https://trends.google.com/trends/explore?q=sentiment%20analysis#qusentiment%20analysis

For hybrid approach, it is a combination of the two previous ones: it uses both lexicons and machine learning algorithms. The earliest work is of [12] how presents a combined classification hierarchy by applying sequentially multiple classifiers. Moreover, [13] use a lexicon of 5244 adjectives, a lexicon of 3296 idioms to improve the classification of sentences made with SVM. [14] apply a hybrid approach to predicting the sentiment strength of an Arabic tweet. In fact, they used a set of linear regression models for predicting initial scores for sentences, then they adjusted these scores by applying a set of rules extracted from existent sentiment lexicon.

Works on Arabic SA are fewer than those on English. The mainly reasons behind that are the followings:

- Limited number of resources developed for ASA: there are few corpora and lexicon freely available [24]. For more details about previous works on ASA, we refer the reader to the extensive surveys presented in [28]. [29] summarizes the list of all freely available SA corpora for MSA and its dialects.
- The *MSA* is a semitic language with rich morphology.
- The diacritization problem of *MSA* (Table 1 shows the meaning change of the word "جمل" /jml/ while changing its diacritics).
- The way of negation detection: the existence of a negation term reverses the polarity.
- The structure of the statement (structered, semi-structered or non structered) has an impact on polarity prediction [27].
- The problem of figurative language: irony, sarcasm, etc.
- The use of foreign words (English, French, Italian, etc) in Internet user's content makes the ASA more difficult.

Word	**Possible Diacritics**	**Transliteration**	**Translation**	**Polarity**
جمل	جُمَلٌ	/joumalun/	sentenses	neutral
	جَمَلٌ	/jamalun/	camel	neutral
	جَمَّلَ	/jammala/	beautify	positive

Table 1. Different meaning for the word "جمل"

Methodology

This work falls within the framework of the machine learning based approach. In fact, many machine learning algorithms require, as input, vector representations. The most common representation used in NLP is the bag of words (BOW) representation.

Despite its popularity, the BOW has two major drawbacks: the lost of the order words and the semantic ignorance of words. Distributed representations resolve these problems. We distinguish mainly two types of embeddings:

- word embeddings: word2vec [21] and Glove [22], etc.

– document embeddings: paragraph vector [15] for variable length texts, sentence vector [23], etc.

Paragraph vector algorithm allows obtaining distributed representations (Doc2vec) for any length sequence, ranging from phrases to documents. It efficiently computes document vector representations in a dimensional vector space. Word vectors are located in the vector space where words that have similar semantic and share common contexts are mapped nearby each other in the space.

The Doc2vec representations were used for English sentiment analysis by [15]. The authors, Le and Mikolov, achieved the best performance with paragraph vector compared to other approaches on IMDB [16] dataset which contains 100000 film reviews. Motivated by their work, we propose using Doc2vec embeddings for Arabic sentiment analysis. The main question asked in this work consists on measuring the efficiency of Le and Mikolov's SA method for Arabic language.

We built a system composed with two parts: the first one applies some linguistic preprocessing on the input text, and the second uses a classifier in order to predict the polarity of the input. We trained two classifiers: a logistic regression LR and a multilayer perceptron MLP [2]. The input vector of the classifier is the embeddings obtained by learning paragraph vector. This vector is a concatenation of the two learned vectors, one from distributed memory version DM and one from distributed bag of words version DBOW, each have 400 dimensions. So that, 800 is the dimension of the classifier's input. In fact, we kept the same neural architecture and the same hyperparameters of paragraph vector model used by Le and Mikolov [15].

Experiments and results

In this section, we perform experiments for two tasks: binary sentiment polarity classification and five-class classification. We test two classifiers: MLP and logistic regression.

Training data and feature extraction

The learning of Doc2vec representations needs a big corpus. According to our knowledge, LABR dataset [17] is the biggest arabic dataset for SA that is freely available[3]. We used the corpus LABR for ASA . This corpus consists of 63257 book reviews written in MSA and colloquial Arabic, each with a rating 1 to 5 stars. Table 2 describes the distribution of the reviews on different classes.

Data preprocessing

We use LABR dataset that contains book reviews. The plain reviews without any preprocessing consists the baseline of our experiments. In other words, each token in the review is considered as a normal word.

For sentiment analysis, some special characters such as ! ? carry sentiments. Moreover,

[2] The MLP contains one hidden layer with 50 units in order to predict the sentiment.

[3] LARB dataset is available on http://www.mohamedaly.info/datasets/labr

	Very negative	Negative	Neutral	Positive	Very positive	Total
Training	2331	4195	9762	15189	19129	50606
Test	608	1090	2439	3865	1649	12651

Table 2. LABR corpus: the reviews distribution on different classes

some combinations of these special characters, for example :) :(, are smileys which are significant for our task. So it is important to consider them as words. Following an analysis of our corpus, we found that many punctuations are agglutinated to words. For this reason, the first preprocess applied over LABR consists on separating punctuations from words and considers them as normal words.

An other experimentation consists on applying stemming for LARB. In fact, the stemming (either light or not) reduces the size of vocabulary. The stemming is the process of eliminating the affixes of words and reducing them to their roots. However, the light stemming removes only prefixes and/or suffixes, without manipulation of the infixes of the word. For example, the two words رَائع et مروع (table 3) have the same stem [rgb]0.24,0.7,0.44or root (ع, ا, ر) but, they don't have the same polarity. So, applying light stemming[4] is relevant for Arabic SA.

Stem	**Light stem**	Transliteration	**Translation**	**Polarity**
(ع, ا, ر)	مروع	/mrwE/	terrible	negative
(ع, ا, ر)	رَائع	/r|}E/	fabulous	positive

Table 3. Light stemming and polarity

Arabic SA experiments

In this work, two types of classification are performed: binary classification and multi-class one. Binary classification considers only two classes: positive and negative. However, in multi-class classification, there are five classes: very positive, positive, neutral, negative and very negative. The same method and hyperparameters are used for both classification tasks: binary sentiment classification and five-classes classification.

To evaluate the performance of SA on the LABR dataset, we carried out several experiments using various configuration. All the experiments were conducted in Python using Theano[5] for classification and gensim[6] for learning vector representation. For machine learning methods, we investigate two classifiers: logistic regression LR and multi-layer perceptron MLP. The input of the each sentiment classifier is a set of features vectors obtained with paragraph vector algorithm. In fact, we tested three different

[4] In this work, we use the light stemmer https://github.com/motazsaad/arabic-light-stemmer

[5] http://deeplearning.net/software/theano/

[6] https://radimrehurek.com/gensim/

types of Doc2vec vectors: (1) vectors obtained with DM version of paragrath vector algorithm, (2) vectors obtained with DBOW version, and (3) concatenation of the vectors obtained separately with DM and DBOW.

Results and discussion

In binary classification framework, the results of the different classifiers with different experimental prepocessing are presented in Table 4. The empty set symbol ∅ means that there is no preprocessing step: we used the review as it stands, without any modification. It represents the baseline of the experiments conducted. The MLP classifier is more efficient than the logistic regression. However, this is not the case when applying preprocessing: the regression classifier becomes more efficient. We notice that there is a little difference in the performances of two classifiers. The lower error rate is 23.31% and it is obtained with logistic regression by applying light stemming. There is a 2% gain after light stemming and special character preprocessing. We think that this low value of gain obtained by applaying light stemming comes from the quantity of MSA words that exist in the corpus. In fact, The reviews of LABR dataset are written in MSA and dialectal Arabic.
We conclude that Arabic language, as opposed to English, requires a specific processing process in order to enhance the performance of SA.

	Regression	MLP
∅	25.60%	24.61%
Special character	25.32%	25.46%
Light stemming	23.31%	23.35%

Table 4. Error rate of various experiments over LABR dataset

It's well known that paragraph vector can capture the semantic similarity between words. Or, among the objectives of this paper is to measure the effeciency of document embeddings for ASA. For example, the words جيد "good" and مُمتَاز "excellent" are close to each other.
To ensure the effectiveness of Doc2vec algorithm for arabic language, we look to the most similar words to some sentiment words. Here, we report the 10 top words similar to the word جيد "good" which are in the following order: جميل beautiful, رَائع fabulous, مُمتع enjoyable, مفيد useful, شيق interesting, مُمل boring, خفيف light, مُمتَاز excellent, لطيف nice , جدًّا very. Among these words, seven words are semantically similar to جيد "good". We notice some similarity errors:

- The word مُمل "boring" is close to جيد "good", which is not true.

- The word مُمل "boring" is closer to جيد "good" than مُمتَاز "excellent", which is false.

We think that this similarity error is strongly linked to the way of Doc2vec learning. In fact, paragraph vector algorithm extract representations that covers syntactic and semantic information based on the context. This means that words with similar context are very near in the vector space, even antonyms. To circumvent this problem, the representations should be constructed by predicting the context and the polarity at a time.

	Regression	**MLP**
Error rate	67.62%	69.42%

Table 5. Error rate in a multi-class classification framework

For multi-class classification, we tested also MLP and LR classifiers. The performances obtained in multi-class classification framework are reported in table 5. In this framework, the input of each classifier is LABR dataset after application of light stemming and special character preprocessing. Table 5 shows that logistic regression is more efficient that MLP. In fact, the error rate with regression is lower than with MLP. Moreover, the error rate in the binary classification framework is lower than in multi-class framework. Indeed and under the same dataset preprocessing and classifier hyperparameters, the error rate obtained with regression is 23.31% with binary classification and 67.62% in multi-class classification which is obviously much harder to handle. In fact, having more classes is not the only challenge imposed by multi-class classification. The other difficulty comes from the relation between some classes, i.e the relation between positive and very positive polarities and relation between negative and very negative polarities.

works with LABR	**Accuracy**
Our work	32.38%
[10]	45%
[11]	45.7%

Table 6. Flat hierarchy for multi-class classification

In this work, we adopted a flat classification (table 6) and we obtained an accuracy equal to 32.38% by using regression classifier over Doc2vec representations. However, [10] used muti-Naive Bayes over BOW vectors. They obtained 45% as accuracy. All works mentioned in tables 6 and 7 use LABR dataset for their experiments.

Works with LABR	**#Levels**	**Accuracy**
[11]	2	46.2%
	4	57.8%

Table 7. multi-level hierarchy for multi-class classification

[11] prove that multi-level hierarchy enhance the performance of multi-class framework (table 7). They used KNN classifier and they obtained an accuracy equal to 46.2% with 2-level hierarchy. But, they obtained 57.8% as accuracy with a 4-level hierarchy.

Conclusion and future works

In this paper, we have made an Arabic sentiment analysis which uses embedding. The aim of this study is to measure the utility of Doc2vec embeddings in Arabic SA framework. The results reported in this paper match the difficulty of Arabic with respect to English. Arabic is morphological rich language. So dealing with Arabic requires preprocessing step. With the purpose to study the potential of preprocessing, we had principally tested the contribution of light stemming in improving performance.

As future work, we think that using tokenization in preprocessing could enhance the performance [30]. Moreover, Adding a stop word list consists an other way of preprocessing. We would like also to test the common BOW representation: the input of our classifiers becomes BOW vectors, not Doc2vec embeddings. So that we could compare the two different representations.

References

1. Pang, B., Lee, L.: Opinion Mining and Sentiment Analysis. In: Foundations and Trends in Information Retrieval (2008)
2. Korayem, M., Crandall, D., Abdul-Mageed, M.: Subjectivity and sentiment analysis for arabic: A survey. In: Advanced Machine Learning Technologies and Applications, vol. 322, pp. 128–139. (2012)
3. Farra, N., Challita, E., Assi, R.A., Hajji, H.: Sentence-level and document-level sentiment mining for arabic texts. In: ICDMW (2010)
4. Almas, Y., Ahmad, K.: A note on extracting sentiments in financial news in english, arabic & ardu. In: CAASL (2007)
5. Abdulla, N.A., Ahmed, N.A., Shehab, M.A., Al-Ayyoub, M., Al-Kabi, M.N.: Towards Improving he Lexicon-Based Approach for Arabic Sentiment Analysis. In: International Journal of Information Technology and Web Engineering, pp. 55–71. (2014)
6. Al-Kabi, M.N., Gigieh, A.H., Alsmadi, I.M., Wahsheh, H.A.: Opinion Mining and Analysis for Arabic Language. In: International Journal of Advanced Computer Science and Applications, pp. 181–195. (2014)
7. Abdulla, N.A., Al-Ayyoub, M., Al-Kabi, M.N.: An extended analytical study of arabic sentiments. In: International Journal of Big Data Intelligence, vol. 1, pp. 103–113. (2014)
8. Bayoudhi, A., Hadrich Belghith, L., Ghorbal, H.: Sentiment Classification of Arabic Documents: Experiments with multi-type features and ensemble algorithm. In: 29th Pacific Asia Conference on Language, Information and computation, Shanghai (2015)
9. Mountassir, A., Benbrahim, H., Berrada, I.: Sentiment classification on Arabic corpora: A preliminary cross-study. In: Document Numérique, vol. 16, pp. 73–96. (2013)
10. Al Shboul, B., Al-Ayyoub, M., Jararweh, Y.: Multi-Way Sentiment Classification of Arabic Reviews. In: the 6th International Conference on Information and Communication Systems (ICICS 2015) (2015)
11. Al-Ayyoub, M., Nuseir, A., Kanaan, G., Al-Shalabi, R.: Hierarchical Classifiers for Multi-Way Sentiment Analysis of Arabic Reviews. In: International Journal of Advanced Computer Science and Application, vol. 7 (2016)
12. El-Halees, A.: Arabic opinion mining using combined models. In International Arab Conference On Information Technology, (2011)
13. Ibrahim, H.S., Abdou, S.M., Gheith, M.: Sentiment analysis for modern standard Arabic and colloquial. In: International Journal on Natural Language Computing (IJNLC), vol. 4 (2015)

14. Refaee, E., Rieser, V.: iLab-Edinburgh at SemEval-2016 Task 7: A Hybrid Approach for Determining Sentiment Intensity of Arabic Twitter Phrases. In: Proceedings of SemEval-2016, pp. 474–480 (2016)
15. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: proceedings ofthe 31 International Conference om Machine Learning (2014)
16. Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (2011)
17. Mahmoud, N., Aly, M., Atiya, A.: LABR 2.0: Large Scale Arabic Sentiment Analysis Benchmark. In: arXiv e-print (arXiv:1411.6718). (2014)
18. Rushdi-Saleh, M., Martin-Valdivia, M.T., Urena-Lopez, L.A., Perea-Ortega, J.M.: Bilingual Experiments with an Arabic-English Corpus for Opining Mining. In: Proceedings of Recent Advances in Natural Language Processing, pp. 740–745, (2011)
19. Biltawi, M., Etaiwi, W., Tedmori, S., Hudaib, A., Awajan, A.: Sentiment Classification Techniques for Arabic Language: A survey. In: the 7th International Conference on Information and Computation System ICICS (2016)
20. Korayem, M.: Sentiment/Subjectivity analysis survey for languages other than English. In: arXiv:1601.00087v2 [cs.CL] (2016)
21. Mikolov, T., chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. (2013)
22. Pennington, J., Rocher, R., Manning, C.D.: Glove: Global vectors for words representation. In: EMNLP, vol. 14, pp. 1532–1543. (2014)
23. Socher, R., Perelygin, A., Wu, J.Y., Chuang, J., Manning, C.D., Ng, A.Y., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Conference on Emperical Methods in Natural Language Processing (2011)
24. Al-Kabi, M., Al-Ayyoub, M., Alsmadi, I., Wahsheh, H.: A Prototype for a Standard Arabic Sentiment Analysis Corpus. In: International Arab Journal of Information Technology, pp. 163–170. (2016)
25. Bayoudhi, A., Ghorbel, H., Koubaa, H., Hadrich Belguith, L.: entiment classification at discourse segment level: Experiments on multi-domain arabic corpus. In: Journal for Language Technology and Computational Linguistics (2015)
26. Al-Ayyoub, M., Bani Essa, S., Alsmadi, I.T.: Lexicon-based sentiment analysis of arabic tweets. In: International Journal of Social Network Mining, pp. 101–114 (2015)
27. Doaa Mohey El-Din Mohamed, H.: A survey on sentiment analysis challenges. In: Journal of King Saud University–Engineering Sciences (2016)
28. Biltawi, M., Etaiwi, W., Tedmori, S., Hudaib, A., Awajan, A.: Sentiment classification techniques for arabic language: A survey. (2016)
29. Mdhaffar, S., Bougares, F., Estève, Y., Hadrich Belguith, L.: Sentiment Analysis of Tunisian Dialect: Linguistic Resources and Experiments. In: Proceedings of The Third Arabic Natural Language Processing Workshop (WANLP), pp. 55–61 (2017)
30. Duwairi, R.M., Qarqaz, I,: Arabic Sentiment Analysis using Supervised Classification. In: The 1st International Workshop on Social Networks Analysis, Management and Security (SNAMS-2014),Barcelona, Spain, (2014)