Sentence-Level Arabic Sentiment Analysis

Amira Shoukry, Ahmed Rafea
Department of Computer Science and Engineering
The American University in Cairo
Cairo, Egypt
am_magdy@aucegypt.edu, rafea@aucegypt.edu

***Abstract* – Arabic sentiment analysis research existing currently is very limited. While sentiment analysis has many applications in English, the Arabic language is still recognizing its early steps in this field. In this paper, we show an application on Arabic sentiment analysis by implementing a sentiment classification for Arabic tweets. The retrieved tweets are analyzed to provide their sentiments polarity (positive, or negative). Since, this data is collected from the social network Twitter; it has its importance for the Middle East region, which mostly speaks Arabic.**

Keywords-component; Sentiment; Feature; Tweets; Polarity

I. Introduction

Sentiment analysis or opinion mining has been currently considered to be one of the most emerging research fields caused by the great opinionated web contents coming from blogs and social network websites. Sentiment analysis is the task of identifying positive and negative opinions, emotions, and evaluations. In general, sentiment analysis aims to determine the attitude of a writer with respect to some topic or the overall tonality of a document [4]. In this study, we are interested in sentiment classification in the Arabic language at the sentence level in which the aim is to classify a sentence whether a blog, review, tweet, etc… as holding an overall positive, negative or neutral sentiment with regards to the given target. It should be noted that this work lies in a larger project that will include extracting sentiment topic and other features.

Choosing to work with the Arabic language is due to several factors. First, the complexity of the language has with regards to both the morphology and the structure has created a lot of challenges which resulted in very limited tools currently available for the aim of sentiment and opinion mining [6], while on the other hand, Arabic sentiment analysis is of growing importance due to its already large scale audience. Second, the Arabic language is both challenging and interesting because of its history, the strategic importance of its people, the region they occupy, and its cultural and literary heritage.

There are mainly two approaches for sentiment classification: machine learning (ML) and semantic orientation (SO). The ML approach is typically a supervised approach in which a set of data labeled with its class such as "positive" or "negative" are represented by feature vectors. Then, these vectors are used by the classifier as a training data inferring that a combination of specific features yields a specific class [17] employing one of the supervised categorization algorithm. Examples of categorization algorithms are Support Vector Machine (SVM), Naïve Bayesian Classifier, Maximum Entropy, etc… On the other hand, the SO approach is an unsupervised approach in which a sentiment lexicon is created with each word having its semantic intensity as a number indicating its class. Then, this lexicon is used to extract all sentiment words from the sentence and sum up their polarities to determine if the sentence has an overall positive or negative sentiment in addition to its intensity whether they hold strong or weak intensity [17]. The SO approach is domain-independent, since one lexicon is built for all domains. The approach we have chosen for sentiment classification is the ML approach because we do not have a lexicon for Arabic sentiment word. This approach is based on selecting a set of features to build feature vectors and train a classifier.

The remaining of the paper shows in more details our achieved work in analyzing and extracting sentiments from the Arabic tweets. Section II summaries the related work done in this area, while section III proposes the system architecture and discusses the system implementation details. Section IV describes the experiments conducted and their results. Finally, Section V talks about the challenges, conclusion and future work.

II. RELATED WORK

According to the type of the classes to predict (positive or negative, subjective or objective), and the levels of classification (sentence, phrase, or document level), the processes of sentiment analysis differ with respect to the technique used whether ML, or SO.

The author in [15] determines the class of the sentence using the average semantic orientation of different phrases. The similarity score measure (Pointwise-Mutual Information or PMI) is used to determine the semantic orientation of each phrase by comparing the similarity of the phrase with a set of ideal 7 positive words with its similarity to a set of ideal 7 negative words. Also, the author in [6] determines the class of

978-1-4673-1382-7/12/$31.00 ©2012 IEEE

the sentence using list storing the semantic orientation of some Arabic word roots which are extracted using a stemmer program. In the classification process, the root of each word is extracted using an Arabic stemmer, and then this root is checked against the stored dictionary. If the root is present, its sentiment is extracted as positive, negative, or neutral. Otherwise, the dictionary asks the user to identify the polarity of the word it has not learned, and add its root to the list of learned roots.

On the other hand, the author in [3] determines the class of the opinions in Web forums in multiple languages: English and Arabic using the ML technique which is the SVM employing combination of both syntactic and stylistic features. The syntactic features used for Arabic were N-grams' frequency, word roots' frequency and punctuation marks' occurrences. POS N-grams were only employed for English but not for Arabic. Whereas the stylistic features used included total words, total characters, special-character frequencies, world-length distributions, character length of forum messages, etc [6]. High accuracy (90%) was noticed by combining both syntactic and stylistic features. Also, the author in [16] built an opinion corpus for Arabic using two different ML techniques: SVM and Naïve Bayes (NB) utilizing various N-grams models like (unigrams, bigrams and trigrams) and getting their term frequency as a weighting scheme. By comparing the results of both learning algorithms, it is noticeable that SVM slightly improves on the performance of NB with an improvement between the best accuracy results of both models of 3.43% for SVM.

The sentiment analysis on Twitter data has been recently the interest of several researchers as Twitter becomes more popular. Most of these researches done in this field have used the ML approach to classify the sentiment of the English tweets with almost very limited or rare work performed to classify the sentiment of tweets in any other language like Arabic. For example, the authors in [19] used a supervised K-Nearest Neighbor (KNN) like classifier to classify English tweets with hash-tags and smileys as features. On the other hand, the authors in [20] used the SVM classifiers to classify the sentiment of the tweets in a two-step approach with abstract features. The training data they have used for their system is gathered from the output of three existing Twitter sentiment classification web sites.

III. TOOLS AND METHODS

After reviewing the majority of the work done in the field of sentiment analysis for Arabic at the sentence level, we wanted to propose an approach that differs and improves upon those proposed ones. In this approach the preprocessing of the tweets is different from the preprocessing done in Arabic sentiment analysis as different stop words list will be used, particularly built for the Egyptian dialect. Also, this approach uses different machine learning classifiers and feature sets. The machine learning classifiers used are Naive Bayes (NB), and Support Vector Machines (SVM). The features used are unigrams and bigrams. The classification and the extraction of the features are done in two distinct components, allowing us to easily try different combinations of classifiers and features until we reach the ones yielding the highest accuracy.

Figure 1, summaries the ML process of the sentence's sentiment analysis in the Arabic language using Arabic tweets from the social network website twitter. The process starts by getting the tweets from twitter. Then we will pass by each tweet and label it as positive, or negative. After that the features in each tweet will be extracted and represented in a feature vector. Then, these feature vectors will be used in the training phase of the classifier. We have used the Weka Suite software for the classification process.

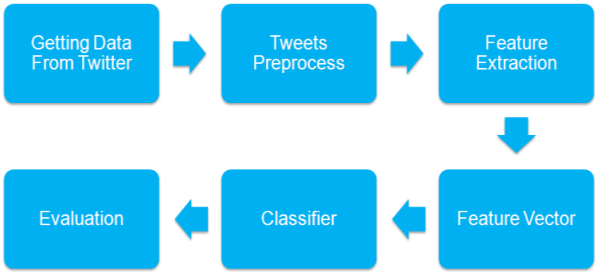

Figure 1. The ML process of the sentence's sentiment analysis

A. *Getting Data from Twitter (Arabic Tweets)*

Although Arabic is considered as one of the top 10 languages most used on the Internet [13], it is considered as a poor content language over the web, unlike English [3] with very few web pages that specialize in Arabic reviews. We have searched for a source that is used to communicate real opinions and at the same time the opinions are written in Arabic. For this reason, we have used Twitter's APIs to get the required tweets, as Twitter provides a search API that lets you search for tweets in a certain language [12]. By setting the language to Arabic: lang=ar, we are now able to get Arabic tweets. Also, it was very important to get a big set of Arabic sentences in order for the classifier to be trained and be able to further classify any new supplied sentence. Twitter was one of the main sources for getting vast amounts of data. We got more than 4000 tweets from twitter from which we have extracted 1000 tweets consisting of 500 positive and 500 negative tweets. We have chosen the tweets that hold only one opinion, not sarcastic, subjective and from different topics. Then we have used these tweets to be our corpus to train the classifier.

B. *Tweets Cleaning and Annotation*

The next step is the process of determining the class of each tweet by annotating it as positive, or negative. Two raters were used to determine the sentiment of the tweets. They had a high degree of agreement in their classification of the tweets, and for those tweets that they disagreed about their sentiment; a third rater was used to determine its final sentiment. Figure 2, shows samples of the annotated tweets. After annotating the tweets, we then went into the process of putting them into a format understandable by the classifier for maximum throughput. This process involved removing the user-names, the pictures, the hash tags, the URLs and all non-Arabic words.

Positive	إنت راجل محترم ...... انا صوتى لن يخرج من دائرة تضمك مع العظيم أبو الفتوح ياريت تتحدوا على رئيس و نائب.....جبهة لا تقهر You are a respectable person … I will definitely vote for you and Abu El Fotoh I hope you can challenge for president and vice president... unbeatable front
Negative	حرام عليك اللى عملته فينا وفى نفسك والله مصر ام الدنيا حرام عليك قوى This is over what you have done to us and to yourself Egypt is the mother of the world this is really over

Figure 2. Samples of positive and negative tweets

C. *Feature Extraction and Feature Vector*

The feature vectors applied to the classifier consisted of the term frequency, as we are using statistical machine learning [7]. First the process starts by extracting all the unigrams and bigrams in the corpus that exceed certain threshold. For example all the unigrams and bigrams with frequencies more than 5 were extracted to be our candidate features. We have chosen to work with unigrams and bigrams as our work is on word/Phrase level sentiment analysis [14]. This can be extended easily using trigrams also. Then for each Tweet, we count the frequency of each candidate features found in it. Thus, for each tweet the following feature vector was constructed using term frequency.

({word1:frequency1, word2:frequency2 …}, "polarity")

For simplicity, in this experiment we have ignored some factors which are to be considered. One of these factors is negation in phrases [10]. In Arabic, they are around 20-words [3]. The negation mechanism in simple terms is inversing the sentence polarity if it is preceded by one of the negation words in phrases. As negation can be local (e.g. not good), or it could involve longer-distance dependencies such as the negation of the proposition (e.g. does not look very good) or the negation of the subject [10]. In general, sentiment analysis seems to require more understanding than the usual topic-based classification.

D. *Weka Suite Software*

The Weka Suite Software version 3.6.43 will be used for the classification process. Weka is written in Java and it provides several ML algorithms such as SVM, NB and others as well as feature selection methods such as IG. It also provides a number of test options, such as cross validation and percentage split. It can be run directly by inserting the dataset into the program or from the command line (when the dataset size is large).

IV. EXPERIMENTATION AND EVALUAION

A. *Results*

The two classifiers: SVM, and NB, were trained first using the frequency of the unigrams only; secondly they were trained using a combination of both unigrams and bigrams with an attempt to capture any negation or sentiment switching phrases. The results were as follows for each classifier using 10-fold validation:

TABLE I. SVM RESULTS

	SVM			
	Accuracy	***Precision***	***Recall***	***F-Measure***
Unigrams	0.721	0.721	0.721	0.721
Unigrams + Bigrams	0.721	0.721	0.721	0.721

a. SVM results before removing stop words

TABLE II. NB RESULTS

	NB			
	Accuracy	***Precision***	***Recall***	***F-Measure***
Unigrams	0.654	0.654	0.654	0.654
Unigrams + Bigrams	0.654	0.654	0.654	0.654

b. NB results before removing stop words

TABLE III. SVM RESULTS

	SVM			
	Accuracy	***Precision***	***Recall***	***F-Measure***
Unigrams	0.726	0.728	0.726	0.725
Unigrams + Bigrams	0.726	0.728	0.726	0.725

c. SVM results after removing stop words

TABLE IV. NB RESULTS AFTER REMOVING STOP WORDS

	NB			
	Accuracy	***Precision***	***Recall***	***F-Measure***
Unigrams	0.652	0.662	0.652	0.646
Unigrams + Bigrams	0.652	0.672	0.652	0.646

d. NB results after removing stop words

Tables I and II show the results obtained in the classification process for the two classifiers: SVM and NB using term frequency scheme respectively before removing the stop words. Tables III and IV show the results obtained using the same techniques but after removing the stop words. Comparing the results obtained to the ones in Tables I and II shows that, there were very small improvements in both SVM and NB. We might explain this behavior as there are no lists available containing the stop words for the Egyptian dialect, thus we have developed this list from the beginning containing all the words that we believe are considered stop words. Given that there were not big improvements in performance, means that there might be some important words that we have removed that should not have been removed, or there are some other stop words that still need to be removed. Thus, we are now in the process of developing a reliable list of stop words that can increase the performance.

B. Discussion

Comparing the results of SVM and NB in both cases, it is clear that SVM has better results than NB. The improvement between the best accuracy results of both models is almost 4-6% for SVM. This behavior was observed in more than one study as usually SVM produces more accurate results than the NB. This is because NB is based on probabilities, thus it is more suitable for inputs with high dimensionality.

Regarding the n-gram model, we can note clearly that bigram model didn't enhance the result using the unigram model. This is because the number of frequent bigrams in the corpus was only 12 bigrams that exceed the threshold of 4. Thus their number was not that effective when extracting the feature vector for each tweet. It should be noted that we have used only the 1000 cleaned and annotated tweets to build the unigram and bigram models, as cleaning is currently done manually.

On the other hand, the results obtained by the SVM have been shown to be highly effective in sentiment analysis outperforming the results obtained by the NB. Because of the principle advantages of the SVM, it was applied successfully in several sentiment analysis tasks. These principle advantages include: "First, they are robust in high dimensional spaces; second, any feature is relevant; third, they are robust when there is a sparse set of samples; and, finally, most text categorization problems are linearly separable" [16]. By comparing the results obtained by SVM in sentiment analysis in general, it is noticeable that SVM overcomes other machine learning techniques.

V. CONCLUSION AND FUTURE WORK

Research in sentiment analysis for the Arabic language has been very limited considered to other languages like English whether at the sentence-level or document-level. In this study, we investigated the ML approach for sentence-level sentiment analysis for Arabic using 1000 tweets from twitter. The results obtained are very promising as a first step.

In our approach, we applied the feature vectors to the NB and SVM Classifiers with the aim of comparing the results and choosing the classifier with the higher accuracy. Problems with regards to the training data is that some tweets may occur many times without any change, through re-tweeting. This gives a misleading boosting to the weight of the terms in the sentence; sometime re-tweets are more than 7 times in the corpus. Also the problem of opinion spamming or untruthful opinions could affect the accuracy of the classification as then the classifier will be built on a misleading tweets. On the other hand, one thing with regards to the testing tweets is that the tweet may contain dual opinions, thus its sentiment to some extend is ambiguous.

For future work, we will continue in this line of research by improving our corpus using techniques such as enlarging or fine-grained annotation. Moreover, we will focus on adding some stylistics features, in addition to considering adding some semantic features thus creating a hybrid approach that combines both the ML and SO approaches. This will be achieved by building a more comprehensive list of all the positive and negative sentiment words for the Egyptian dialect since there doesn't exist any of them. Also, negations and valence shifters will be considered as a feature in ML approach because their presence in the sentence can result in changing the sentiment of the whole tweet like "حلو- nice" implying positive sentiment if preceded by "مش حلو – not good" would then imply negative sentiment. And finally, neutral sentiment tweets has to be considered as in real world applications neutral tweets cannot be ignored.

ACKNOWLEDGMENT

The authors would like to thank ITIDA for sponsoring this project entitled "Semantic Analysis and Opinion Mining for Arabic Web", and the Egyptian industrial company LINK-Development team for their help in developing a tool for collecting and annotating the tweets.

REFERENCES

[1] K. Yessenov, and S. Misailovic, "Sentiment Analysis of Movie Review Comments", Graduation project. 17th, May, 2009.

[2] A. Abbasi, H. Chen, and A. Salem, "Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums" ACM Transactions on Information Systems (TOIS), Vol 26, Issue 3, June 2008.

[3] M. Elhawary, and M. Elfeky, " Mining Arabic Business Reviews", Google Inc., Mountain View, CA, USA, 2010 IEEE International Conference on Data Mining Workshops.

[4] http://en.wikipedia.org/wiki/Sentiment_analysis

[5] T. Helmy, and A. Daud, "Intelligent Agent for Information Extraction from Arabic Text without Machine Translation", Information and Computer Science Department, College of Computer Science and Engineering, King Fahd University of Petroleum and Minerals.

[6] N. Farra, E. Challita, R. Abou Assi, and H. Hajj, "Sentence-level and Document-level Sentiment Mining for Arabic Texts", Department of Electrical and Computer Engineering, American University of Beirut, Beirut, Lebanon.

[7] B. Pang and L. Lee,”Thumbs up? Sentiment Classification using Machine Learning Techniques” Department of Computer Science, Cornell University, Shivakumar Vaithyanathan, IBM Alma den Research Center

[8] Y. Lu, M. Castellanos, U. Dayal, and C. Zhai, “Automatic Construction of a Context-Aware Sentiment Lexicon: An Optimization Approach”, UIUC Computer Science, 201 N. Goodwin Avenue, Intelligent Information Management Lab, HP Laboratories.

[9] A. Farghali and K. Shaalan, “Arabic Natural Language Processing: Challenges and Solutions”, Monterey Institute of International Studies, the British University in Dubai

[10] T. Wilson, J. Wiebe, and P. Hoffmann, ”Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis”, Intelligent Systems Program, Department of Computer Science. University of Pittsburgh,

[11] ZhangWei, “Opinion Mining and Sentiment Analysis: A Survey”, Department of Computer Science, School of Computing, National University of Singapore

[12] Twitter search API, http://search.twitter.com/search.atom?lang=ar&rpp=100&page={0}&q={1}

[13] http://www.internetworldstats.com

[14] L. Khreisata, “A machine learning approach for Arabic text classification using N-gram frequency statistics”, *Journal of Informatics*, Vol 3, Issue 1, January 2009, Pages 72-77.

[15] P. Turney, “Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews”, in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics(ACL) , pp. 417-424, July 2002

[16] M. Rushdi-Saleh, M. Teresa, L. Martín-Valdivia, A. Ureña-López, and J. M. Perea-Ortega, “OCA: Opinion corpus for Arabic”. *Journal of the American Society for Information Science and Technology*, 62: 2045–2054. doi: 10.1002/asi.21598. 2011

[17] S. Morsy, “Recognizing Contextual Valence Shifters in Document-Level Sentiment Classification”. Department of Computer Science and Engineering, The American University in Cairo (AUC). 2011

[18] http://www.cs.waikato.ac.nz/ml/weka

[19] D. Davidiv, O. Tsur and A. Rappoport, “Enhanced Sentiment Learning Using Twitter Hash-tags and Smileys”. Coling 2010.

[20] L. Barbosa and J. Feng, “Robust Sentiment Detection on Twitter from Biased and Noisy Data”. Coling 2010.