Arabic Sentiment Analysis: Lexicon-based and Corpus-based

Nawaf A. Abdulla[1], Nizar A. Ahmed[2], Mohammed A. Shehab[3] and Mahmoud Al-Ayyoub[4]

Computer Science Department
Jordan University of Science and Technology
Irbid, Jordan
[1]naabdulla11@cit.just.edu.jo
[2]naahmad12@cit.just.edu.jo
[3]mashehab12@cit.just.edu.jo
[4]maalshbool@just.edu.jo

***Abstract*— The emergence of the Web 2.0 technology generated a massive amount of raw data by enabling Internet users to post their opinions, reviews, comments on the web. Processing this raw data to extract useful information can be a very challenging task. An example of important information that can be automatically extracted from the users' posts and comments is their opinions on different issues, events, services, products, etc. This problem of Sentiment Analysis (SA) has been studied well on the English language and two main approaches have been devised: corpus-based and lexicon-based. This paper addresses both approaches to SA for the Arabic language. Since there is a limited number of publically available Arabic dataset and Arabic lexicons for SA, this paper starts by building a manually annotated dataset and then takes the reader through the detailed steps of building the lexicon. Experiments are conducted throughout the different stages of this process to observe the improvements gained on the accuracy of the system and compare them to corpus-based approach.**

***Keywords:* Sentiment analysis, Opinion mining, Lexicon-based, Corpus-based, Arabic language.**

I. Introduction

Over the last decade, Internet users started to contribute more to the Internet's contents by adding comments or opinions to the webpages. This is due to the Web 2.0 technology which permits Internet surfers to share their thoughts and views via social networks (such as Twitter and Facebook), personal Blogs, forums, etc. This new technology results in a massive amount of raw data for which creative data mining techniques are needed in order to extract valuable knowledge.

One important objective is to perform Opinion Mining (OM) or Sentiment Analysis (SA), which is to automatically extract users' opinions on certain issues, events, services, products, etc. These opinions are expressed in various forms such articles, reviews, forum posts, short comments, tweets, etc. and the linguistic styles used vary greatly. Thus, extracting the semantic orientation or polarity of these opinions is challenging. The *semantic orientation* or *polarity* is defined as a measure of a word, phrase, sentence, or document's subjectivity and opinion [1]. In other words, it determines whether a sentence or a document is positive or negative.[1] The benefits of performing SA are countless. It can help in measuring the public's opinion on controversial issues in a more accurate, comprehensive and affordable fashion than public polls. It can also help companies tailor their services/product to their customers' needs and thus increase their profit.

According to [2], subjectivity and sentiment analysis studies are classified based on: (I) predicted class (the text is subjective or objective); (II) predicted polarity (be it positive, negative, or neutral); (III) level of classification (SA for a word, phrase, sentence, or a whole document); (IV) the applied approach (supervised or unsupervised). The system proposed in this work deals with subjective texts. It classifies the whole document into one of the three polarity classes (positive, negative and neutral) via both supervised and unsupervised approaches.

In the *supervised* approach, which is also known as the corpus-based approach, machine learning classifiers such as Support Vector Machine (SVM), Naïve Bayes (NB), Decision Tree (D-Tree), K-Nearest Neighbor (KNN), etc., are applied to a manually annotated dataset. The sentiment analysis under this approach can be viewed as a special case of the text classification problem [1]. The dataset is split into a training set and a testing set. The classifier learns from the training data and (sometimes) builds a model which is later used to classify the documents of the testing set. The system's accuracy is determined by measuring the different types of errors made by the classifier. This approach generally achieves a higher accuracy than that of the unsupervised approach for sentiment analysis; however, it requires building a gigantic corpus (dataset) and labeling it manually by human experts. The process of manual annotation can be very difficult even for native speakers due to sarcasm and cultural

[1] Some studies include the neutral case as well.

978-1-4799-2303-8/13/$31.00 ©2013 IEEE

references. It can also be expensive and time-consuming [3]. Moreover, the model built may be a domain-biased. That is, it could give low accuracy when is applied on such a different domain from what it was learned from [4].

On the other hand, the *unsupervised* approach, which is also called lexicon-based approach, states the semantic orientation (polarity) of a word or sentence based on a dictionary. In the dictionary, each word is associated with a simple polarity value (+1, -1 or 0 for positive, negative or neutral, respectively) or perhaps with a polarity "strength" value (e.g., the range (+1 to +5) can be used for positive polarities, where a word with a polarity of +5 is a much more positive word than one with a polarity of +1). The dictionary (lexicon) is constructed either manually or automatically [1]. For the automatic method, an initial dictionary (list of seed words) is given. Then, the dictionary size is increased by employing some "similarity" techniques. It is customary not to list the neutral words in the lexicon since their absence means their polarity is 0. The total polarity of the sentence/document is calculated via extracting the polarity score of each word in the text from the lexicon, and summing their polarities scores into one score that represents the sentiment of the whole text. The lexicon-based approach is very practical though it does not adapt well with different domains [2]. Moreover, its accuracy is often lower than the corpus-based approach.

Some papers (e.g., [4]) suggest combining the two previously mentioned approaches exist through what they call *weakly-* or *semi-supervised* approach. Read and Carroll [4] tweaked the supervised approach via integrating the classifiers with a predefined dictionary. Additionally, they examined the efficacy of word similarity techniques in order to enlarge their lexicon size. The results show that weakly-supervised approach may outperform unsupervised one. Yet, it requires adequate corpus-size to function well.

Generally speaking, the approaches discussed so far are easier to apply on English than Arabic since Arabic, a Semitic language written from right to left, is inflectional and morphological. Its letters can be written with different shapes according to their position in the word. Moreover, Arabic Internet users use a mixture of Modern Standard Arabic (MSA) and local dialects making text processing a very hard and challenging task [5] [10]. Consequently, this complexity propagates throughout all stages in Arabic text classification and sentiment analysis (whether it is supervised or unsupervised) [6].

In this paper, we first collect an Arabic dataset composed of 2000 Tweets; 1000 are positive and 1000 are negative. Next, we build an Arabic lexicon from a seed list of 300 words. Both positive and negative words are listed together with +1 and -1 weights for positive and negative polarities, respectively. After that, we design and implement a lexicon-based tool for SA purposes. Last, as we expand the lexicon, we carry out experimentation to compare the performance of a corpus-based tool and a lexicon-based tool. Since there are rare freely accessible Arabic dataset and lexicons for SA [7], the contributions of this paper of building a dataset and a lexicon are important. This paper takes the reader throughout the detailed steps of building the lexicon. Tools for spelling correction, handling of special words, computing a document's polarity from the polarities of the individual words comprising the document are among the issues and challenges faced when trying to build a lexicon. Additionally, we compare the lexicon-based approach results to the corpus-based ones using the same dataset. The comparison provides us with some insights that would be discussed in later sections.

The paper is ordered as follows: Section II provides a glance about what other studies in SA have done; specifically what other researches on Arabic SA have conducted. In Section III, we explain our approach: dataset description, lexicon construction and tool design and implementation. Then, we investigate the results for both corpus-based and lexicon-based, and discuss the differences and similarities in Section IV. Finally, conclusion and future work are outlined in Section V.

II. RELATED WORK

Many studies have been conducted in the sentiment analysis field. Researchers have proposed interesting approaches and developed various systems to deal with this problem. Unfortunately, most of these systems are developed for the English language and are not directly usable on other languages. In this section, we discuss several key papers in this field and try to highlight the works useful for the Arabic language.

Melville et al. [8] tried to deal with supervised sentiment analysis. They took the advantages of building a model that depend on a lexicon and used a model that depend on labeled documents. They combined these two models in order to have a multinomial Naïve Bayes classifier.

Go et al. [9] used many machine learning classifiers such as NB, maximum entropy and SVM. They utilized tweets collected and labeled manually, and inserted emoticons. These emoticons were used as noisy labels.

Abdul-Mageed et al. [10], have proposed a system called SAMAR for Subjectivity and Sentiment Analysis (SSA), which requires identifying whether the text is objective or subjective before identifying its polarity. The proposed system uses the SVM^{light} algorithm for classification and the dataset they used was collected from four different genres of social media websites: chat, Twitter, Web forums and Wikipedia Talk Pages. Their experiments showed how difficult and complex the characteristics of Arabic language in SSA [10].

Hu et al. [11] used unsupervised sentiment analysis. They depended on emotional signals. The emotional signals were divided into two categories: emotion correlation and emotion indication. They conducted their approach on two Twitter datasets. The experimentation outcomes indicated the efficiency of the proposed methodology and the important of the roles of different emoticons in sentiment analysis as well.

In [12], Dasgupta et al. used clustering technique, where text is clustered into sentiment dimension. This happens by providing user feedback in spectral clustering process in an interactive manner. Each dimension of spectral clustering contains features. These features can be considered as sentiment oriented topics. In order to determine the most important dimensions, they needed human interaction. Thus, this proposed framework needs a massive amount of manual annotation for documents. This study requires the user to select the features for dimension expansion; that is, the user must have previous knowledge to perform such a task.

An Arabic dataset consisting of 500 movie reviews was built by Rushdi-Saleh et al. [13]. Similar to [9], the authors used SVM and NB in their study. They started their study by preprocessing the collected dataset. The conducted preprocessing operations included manual spelling correction, stop-words removal, stemming, and N-Gram tokenization. Although their experimentation results showed accuracy close to 89%, the size of the dataset they used was small compared to other datasets used in other English-based studies. On the other hand, Al-Subaihin et al. [14] have created and implemented a lexicon-based sentiment analysis tool for colloquial Arabic text. They applied it on a dataset comprised of Arabic forums comments and newspaper articles written in Arabic.

Finally, Shoukry and Refae in [15] worked on a tweet dataset composed of 1000 tweets (500 are positives and 500 are negative). They dealt with sentence-level sentiment analysis since tweets length is restricted to 140 characters. Though their work lacks handling the neutral cases and exploits a small corpus, they explored the direction of Arabic dialects and appended some words from the Egyptian dialect alongside the MSA ones. For the preprocessing phase, they applied Unigram-based and Bigram-based features extraction techniques and concluded that there is no difference in the results. The approach followed in this paper was corpus-based (supervised approach), where SVM and NB were used for polarity classification. The results showed that SVM outperformed NB in sentiment analysis with an accuracy of 72.6% regardless to the feature extraction technique used (whether it is Unigram-based or Bigram-based).

III. Approach

In this section, we discuss both approaches for SA: corpus-based and lexicon-based. The first step (discussed in the following subsection) is collecting and annotating a dataset, a very expensive step in terms of both time and effort. The dataset is used for building a classification model for the corpus-based tool as well as testing the lexicon-based tool. The next set of challenges is related to building the lexicon. We discuss these challenges and the tools/approaches used to handle them in the following subsections.

A. Dataset and Pre-processing

The type of documents comprising the dataset (whether they are comparative essays, reviews, comments, tweets, etc.) can have a serious effect on SA since it governs the document's length and the language used (how formal is it, the cultural references used, whether sarcasm is used, whether it is a single language or a mixture of several languages, etc.). We decided to work with tweets due to the popularity of social networking websites and the ability to extract a lot of useful information from what people post on such websites [16]. By using a tweet crawler, we have collected 2000 labeled tweets (1000 positive tweets and 1000 negative ones) on various topics such as: politics and arts. These tweets include opinions written in both Modern Standard Arabic (MSA) and the Jordanian dialect, as shown in Fig. 1.

Positive	انا احب هذا الكاتب I like this author
Negative	الله يكون في عون الفقير و الطبقه المتوسط سوف تنحدر اكثر و اكثر God help the poor and the middle class will diminish more and more

Fig. 1 Sample of positive and negative tweets

The tweets collected must contain some kind of feelings and it is the objective of our system to extract valuable information from such tweets in order to determine the semantic orientation of the text of the tweet. Table I depicts some statistics on the collected dataset that might attach valuable information to the results later on. The annotation process of the tweets was manually conducted by two human experts. If both experts agree on the label of a certain tweet, then the tweet is assigned to this label. Otherwise, a third expert is consulted to break the tie.

TABLE I
STATISTICS ON THE TWEETS DATASET

	Positive	**Negative**
Total tweets	1000	1000
Total words	7189	9769
Avg. words in each tweet	7.19	9.97
Avg. characters in each tweet	40.04	59.02

After building the dataset, some pre-processing techniques are automatically applied to it. First of all, we saved each tweet into a single file and split all tweets either in a positive folder or a negative one. Then, our tool automatically corrects misspellings and removes the repeated letters (e.g., muuuuuuch = كثيييير). We use the MS Word dictionary [17] as a reference for misspelling correction and select the first word suggested by it automatically. As for repeated letters, we used a naive algorithm which simply counts the number of letters in each word. If the number exceeds 5, then it eliminates the repeated letters and looks it up in the MS Word dictionary.

Next, all stop-words are removed. A list of Arabic stop-words was obtained from the Khoja stemmer tool [18] and we added some extra stop-words from different Arabic dialects. After that, a normalization process for the letters (أ and ة) is done by our tool; that is, all shapes of the letter "alif" (e.g., أ إ ا) are normalized to (ا) and the letter "ta'a" (ة) is converted into (ه). The reason for applying such a process is that many Arabic Internet users often mistake between these similar letters and use them interchangeably.

B. Lexicon Construction

Due to its complex nature, we have dedicated most of our efforts for building and enhancing the lexicon manually. We started with 300 seed words taken from SentiStrength website [19]. These words are first translated into Arabic using English–Arabic dictionary. Positive words have +1 polarity while negative ones have -1. We combined them together into one lexicon in order to reduce the searching time for our tool.

To improve the performance of the lexicon-based tool, several extensions are applied to the lexicon. E.g., we add the synonyms of each word and gave them the same polarity as their original word. For each word, 2 to 3 synonyms (on average) are added. In addition to the ordinary words that any other lexicon can include, we append emoticons {:), ;), :-(} that define a sentiment.

Finally, the resultant lexicon has been expanded from 300 words to 3479 words consisting of 1262 positive words and 2217 negative ones. Any word that is not included in the lexicon will be considered neutral with a zero weight.

C. Tool Design and Implementation

Several algorithms and tools have been designed to deal with unsupervised sentiment analysis. For example, the authors of [20] discuss the English version of the SentiStrength program [19], an unsupervised tool for extracting the polarity and orientation of unstructured text documents. Many researchers [1, 20, and 21] use a lexicon-based technique for sentiment analysis to avoid the expensive process of manually annotating a dataset. Accordingly, in our approach, we build a tool to find the sentiment orientation of an Arabic text, where each word in the lexicon has a weight representing the polarity (+1 and -1 for positive and negative words, respectively). If the word is inexistent in the lexicon, its polarity is zero. Then, for feature extraction, we use the Unigram technique; that is, each word in the tweet (text) is considered as one token regardless to the surrounding words. Finally, after extracting the polarity of each word from the Arabic lexicon, the tool aggregates the total weights and generates the polarity of the entire inputted text. The following equation is used in our tool [22]:

$$\sum_{i=0}^{m} wi$$

Where m is the total number of words in the text, and wi is the weight or the polarity of each scanned word. Below we depict the algorithm used to compute the polarity of a text document.

It is worth mentioning that our tool can handle both negation and intensification which most proposed Arabic tools in this field somehow avoided [7, 15]. Negation refers to words that reverse the polarity (sentiment) of the word coming after them. The following is the list for the main negation words in MSA: {"لا", "لم", "لمّا", "لن", "ما", "ليس"}. For example, "لا أحب الحلوى" which means "I don't like sweets". In this example, "أحب" ("like") is positive word in our lexicon with polarity equals +1. Yet, due to the negation word "لا" ("don't") that came before it, the polarity turns to -1 (this is called the switch negation [1]). In other words, we switch the polarity of any word into completely the opposite. Also, we enrich the list by inserting some negation words from different dialects such as "مش", "مو", "غير", etc.

INPUT: Text File (comment or review) $\mathcal{T}$, The sentiment lexicon $\mathcal{L}$.
OUTPUT: S_{mt} = {***P***, ***Ng***, or ***Nt***}, *where* ***P****: Positive,* ***Ng****: Negative,* ***Nt****: Neutral.*
INITIALIZATION: Sum = 0, *where sum: accumulates the polarity of all tokens* $t_{i\text{-}Smt}$ in $\mathcal{T}$.
Begin

```
For each t_i ∈ T do
    Search for t_i in L
  If t_i ∈ L then
      Sum ← Sum + t_i-Smt
  End If
End For
If Sum > 0 then
    S_mt = P
Else If Sum < 0 then
    S_mt = Ng
Else
    S_mt = Nt
End If
```

End

On the other hand, intensification (some studies refer to it as booster words) increases the intensity of a word's polarity to make it even stronger. According to an Arabic linguist, the main booster words in the Arabic language can be narrowed into these four words: {"جدا", "تماما", "فعلا", "بإفراط"}. Our approach deals with such cases by adding +1 to a positive word and -1 to a negative one [1, 5]. For instance, the sentence "أنت رائع جداً", which means "you are very cool", has a polarity equals +2 since "رائع" ("cool") has +1 in our lexicon, and the appearance of "جداً" ("very") intensifies the polarity by adding another +1. However, not only does a booster word come after an adjective in Arabic language, but also it might come before it (e.g., "أنت جداً رائع", which is written in the form "you are cool very"). Our tool can handle such cases as well.

IV. Results

The objective of the experiments is to compare the accuracy of the two approaches for SA (corpus-based and lexicon-based) and show how the accuracy of the lexicon-based tool improves with the addition of more words to the lexicon. To evaluate the accuracy of each tool, we rely on three of the most widely used accuracy measures in the literature [7, 13, and 15] which are *precision*, *recall*, and *accuracy*. Their mathematical equations are listed below.

$$\text{Precision} = TP / (TP + FP)$$

$$\text{Recall} = TP / (TP + FN)$$

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

Where TP, TN, FP and FN are true positive, true negative, false positive and false negative, respectively. These metrics are used for both supervised and unsupervised experiments.

A. Supervised Experiments

The aim of this part of experimentations is to focus on corpus-based approach for SA. We use the RapidMiner software [23] to carry out the experimentation. This tool is built for data mining and machine learning purposes and contains a built-in text pre-processing tasks such as tokenization, stop words removal, weighting schemes, stemming, etc. Three different experiments are conducted using four famous classifiers: SVM, NB, D-Tree, and KNN where K=9 because it gave us the best accuracy than K= 1, 2, 3 and so forth. The selection of these classifiers is motivated by the superior performance they showed in previous related studies. For testing and validation purposes, we applied the 5-fold cross validation technique instead of 10-fold due to the hardware limitations (out-of-memory error). Finally, we test the effect of different stemming techniques (root-stemming, light-stemming and no stemming) on the performance of each classifier. Table II shows the experiment's outcomes, which clearly confirms the well known result that SVM and NB have better accuracy than other classifiers when it comes to text classification. Table II also shows that light-stemming gives better results than other stemming techniques.

TABLE II
SUPERVISED SENTIMENT ANALYSIS RESULTS

Classifiers	Accuracy		
	Original	**Root-stemmer**	**Light-stemmer**
SVM	84.7%	85%	87.2%
NB	80.4%	78.75%	81.3%
KNN	51.3%	52.85%	51.45%
D-tree	50%	50%	50%

B. Unsupervised Experiments

We apply our proposed lexicon-based tool on the 2000 collected dataset after being root-stemmed and the results show that lexicon-based approach gives much lower accuracy compared with the corpus-based. Table III illustrates the precision, recall, and accuracy obtained from the experimentations.

We attempted to build the lexicon gradually and conduct the experiments over three phases during the lexicon construction. In the first phase, the lexicon is at its smallest size with only the original words that were taken from the SentiStrength website. Since we also took the stems of the available words in the website, their number is raised to almost 1000 words. The second phase is after appending some synonyms of the 300 original stemmed words. So, the size of the lexicon increased to reach around 2500 words. The last phase involved adding some random samples into the lexicon (positive words and negative ones). The final lexicon size is expanded to 3479 words.

The outcomes of these three experiments illustrates that the bigger the lexicon is, the better the results are. This is obvious in Table III where the accuracy of the tool proportionally increases with the lexicon size. Nonetheless, it could be noticed that the accuracies gab reduces when the lexicons reach large sizes. In other words, the accuracy difference between phase I (about 1000 words) and phase II (about 2500 words) is approximately 30%, while the gap between phases II and III (the lexicon in phase III has almost 1000 additional words) is around 10%. In conclusion, we can speculate that the amount of work (i.e., the size of the lexicon) required for each increase in the accuracy of the lexicon-based tool grows rapidly. This means that it might not always be the best choice to expand the lexicon in order to improve the accuracy. Better processing and smarter ways of computing the polarity of a text (based on the polarities of individual) might also have significant effect on the accuracy. We elaborate more on these ideas in the following paragraphs.

TABLE III
LEXICONS' SCALABILITY RESULTS

Evaluation Metrics	Phase I	Phase II	Phase III
Precision	10.5%	52.2%	58.6%
Recall	8.9%	48.9%	64.9%
Accuracy	16.5%	48.8%	59.6%

Based on our reading of the related works in the literature, we expected such modest performance of the lexicon-based tool. The experimentations revealed more insights into why such tools perform badly. The first and most obvious issue is the lexicon size, which was not big enough. 3479 of positive and negative words do not cover as many sentiments as required. Hence, we had better seek for other techniques to build the lexicon along with the manual one. Approaches like building lexicon automatically or using weakly-supervised techniques [4] to enlarge the lexicon can save time and effort.

Predicted Positive	عن جد هالحكومة عاطينا حريتنا بالكاااااااااامل هههههههه This government really gives us full freedom hahaha
Predicted Positive	عن جد دمك خفيف You are really hilarious

Fig. 2 Examples of sarcastic tweets

Secondly, the presence of sarcastic comments/tweets really could impact the tool's results badly. These comments/tweets have apparently positive sentiments, but they are implicitly very ironic and negative, as shown in Fig. 2. So, as a primitive solution, we may reconsider the type of comment/tweet in the future and exclude sarcastic ones. By studying them independently and developing separate tools/techniques to automatically detect sarcasm and handle it properly within the context of SA, we can greatly improve the performance of lexicon-based tools.

V. CONCLUSION AND FUTURE WORK

This paper addresses both approaches to Sentiment Analysis (SA): corpus-based and lexicon-based. Because the publically available Arabic dataset and lexicons for SA are rare and limited, this paper discusses building a manually annotated dataset and then takes the reader through the detailed steps of building the lexicon. Experiments are conducted throughout the different stages of this process to observe the improvements gained on the accuracy of the system. It is observed that the corpus-based tool that uses SVM for classification of a light-stemmed dataset gives the highest accuracy. Moreover, it is perceived that with the increase of the lexicon, the accuracy of the lexicon-based tool improves. While these results are known/expected, another observation is worth thinking: the amount of work (i.e., the size of the lexicon) required for each increase in the accuracy of the lexicon-based tool grows rapidly.

This study represents the baseline for our future work, where we plan to enlarge the dataset along with adding a third polarity case (neutral class). Additionally, augmenting the lexicon size will provide much better results, especially when adding strength to it- the polarity of words could vary between -5 to +5, which may end up with more accurate outcomes. One thing more, the sarcasm in some tweets always leads to misinterpretation and consequently a wrong polarity classification. Hence, taking into consideration sarcasm absolutely will results in higher accuracy.

REFERENCES

[1] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational linguistics* 37, no. 2, pp. 267-307, 2011.

[2] M. Korayem, D. Crandall, and M. Abdul-Mageed, "Subjectivity and Sentiment Analysis of Arabic: A Survey," In *Advanced Machine Learning Technologies and Applications*, pp. 128-139. Springer Berlin Heidelberg, 2012.

[3] Y. He, and D. Zhou, "Self-training from labeled features for sentiment analysis," *Information Processing & Management* 47, no. 4, pp. 606-616, 2011.

[4] J. Read, and J. Carroll, "Weakly supervised techniques for domain-independent sentiment classification," In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pp. 45-52. ACM, 2009.

[5] A. Farghaly and K. Shaalan, "Arabic natural language processing: Challenges and solutions," *ACM Transactions on Asian Language Information Processing (TALIP)* 8, no. 4, 2009: 14.

[6] A. El-Halees, "Arabic text classification using maximum entropy," *The Islamic University Journal (Series of Natural Studies and Engineering) Vol.* 15, pp. 157-167, 2007.

[7] M. Al-Kabi, A. Gigieh, I. Alsmadi, H. Wahsheh, and M. Haidar, "An Opinion Analysis Tool for Colloquial and Standard Arabic," In *The fourth International Conference on Information and Communication Systems (ICICS 2013)*, 2013.

[8] P. Melville, W. Gryc, and R. D. Lawrence, "Sentiment analysis of blogs by combining lexical knowledge with text classification," In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1275-1284, ACM, 2009.

[9] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Project Report, Stanford,* 2009, pp. 1-12.

[10] Abdul-Mageed, Muhammad, Sandra Kübler, and Mona Diab. "Samar: A system for subjectivity and sentiment analysis of arabic social media." In Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis, pp. 19-28. Association for Computational Linguistics, 2012.

[11] X. Hu, J. Tang, H. Gao, and H. Liu, "Unsupervised Sentiment Analysis with Emotional Signals," Rio de Janeiro, Brazil ACM 978-1-4503-2035-1/13/05, 2013.

[12] S. Dasgupta, and Vincent Ng, "Topic-wise, sentiment-wise, or otherwise?: Identifying the hidden dimension for unsupervised text classification," In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pp. 580-589. Association for Computational Linguistics, 2009.

[13] M. Rushdi-Saleh, M. T. Martín-Valdivia, L. A. Ureña-López, and J. M. Perea-Ortega, "OCA: Opinion corpus for Arabic," *Journal of the American Society for Information Science and Technology* 62, no. 10 pp. 2045-2054, 2011.

[14] F. Sebastiani, "Machine learning in automated text categorization," *ACM computing surveys (CSUR)* 34, no. 1, pp. 1-47, 2002.

[15] A. Shoukry, and Ahmed Rafea, "Sentence-level Arabic sentiment analysis," In *Collaboration Technologies and Systems (CTS), 2012 International Conference on*, pp. 546-550, IEEE, 2012.

[16] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," In *Proceedings of LREC*, vol. 2010, 2010.

[17] Microsoft Office. (2013) The Office website. [Online]. Available: http://office.microsoft.com/en-GB/?CTT=97

[18] Shereen Khoja. (1999) The Pacific University homepage. [Online]. Available: http://zeus.cs.pacificu.edu/shereen/research.htm

[19] The SentiStrength website. [Online]. Available: http://sentistrength.wlv.ac.uk/SentStrength_Data/

[20] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social web," *Journal of the American Society for Information Science and Technology* 63, no. 1, pp.163-173, 2012.

[21] X. Ding, B. Liu, and P. S. Yu, "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.

[22] J. A. Bullinaria, and J. P. Levy, "Extracting semantic representations from word co-occurrence statistics: A computational study," *Behavior Research Methods* 39, no. 3, pp. 510-526, 2007.

[23] The rapid-I website. [Online]. Available: http://www.rapid-i.com/