Sentiment Analysis of Arabic Tweets Using Semantic Resources
Lamia Al-Horaibi and Muhammad Badruddin Khan
Pages 9 – 18
DOI: http://dx.doi.org/10.21700/ijcis.2017.xxx

Sentiment Analysis of Arabic Tweets Using Semantic Resources

Lamia Al-Horaibi Muhammad Badruddin Khan
Department of Information Systems, Al Imam Mohammad Ibn Saud Islamic University (IMSIU)
SAUDI ARABIA
lamo4e@hotmail.com, badruddin@ccis.imamu.edu.sa

Abstract: *Sentiment analysis has grown to be one of the most active research areas in natural language processing and text mining. Many researchers have investigated sentiment analysis and opinion mining from different classification approaches. However, limited research is conducted on Arabic sentiment analysis as compared to the English language. In this paper, we have proposed and implemented a technique for Twitter Arabic sentiment analysis consisting of a semantic approach and Arabic linguistic features. Hence, we introduced a mechanism for preprocessing Arabic tweets, and for the methodology of sentiment classification we used a semantic approach. Also, we proposed a technique of classification which uses both Arabic and English sentiment lexicons to classify the Arabic tweets into three sentiment categories (positive or negative or neutral). Our experiments show that many issues were encountered when we used the Arabic SentiWordNet facility to classify Arabic tweets directly; these issues are basically related to Arabic text processing. The Arabic lexicons and Arabic tools must be improved or built from scratch in order to improve Arabic sentiment analysis using the semantic approach. The improvement in results, which are due to our contribution in the form of enhanced Arabic lexicons and amended Arabic tools, demonstrate this need.*

Keywords: *Arabic Lexicons, Sentiment Analysis, Semantic Approach, SentiWordNet, Twitter.*

Received: *June 9, 2016* | **Revised**: *November 10, 2016* | **Accepted**: *January 21, 2017*

1. Introduction

Since 2000, there has been a rapid increase in the written contributions of people via social media platforms, especially Twitter and Facebook. This trend has resulted in the production of a large amount of data on the web. The amount of data has encouraged and attracted researchers to create and use efficient methods to handle and analyze the variety of types and formats of textual data for a broad range of languages [1]. Furthermore, the data that are expressed on social media in the form of opinions or reviews or posts constitute an interesting area worthy of exploration. There is a new trend in companies, organizations and governments to monitor public sentiment expressed on social networks because it is considered a valuable source of information. This information can be valuable for studying a user's behavior and opinions on products, events, issues, news, games, and so on. The objective of sentiment analysis is to determine opinions, feelings, emotions, beliefs, and attitudes, whether positively, negatively, or neutrally expressed in their own language with reference to a person, an organization, a product, a location or an event [3].

The Twitter platform is a very attractive source for sentiment analysis. This is because it allows users to post and share real-time messages about their attitudes towards different topics, to discuss and comment on a variety of issues, and express their opinions on products they have used [4]. Twitter is considered one of the popular social networks in the Arab region. The latest statistical reports produced by the Dubai school of government[1] indicate that there were over 5.7 million Arab users on Twitter in March 2014. Furthermore, around 17 million tweets per day are posted by Arab users. The country with the highest number of active Twitter users in the Arab region is Saudi Arabia, with 2.4 million users. Saudi Arabian users post 40% of all the tweets in the Arab world, while Egyptian users post 17% and Kuwaiti users post 10%.

However, increasing the amount of information available in Arabic languages makes the task of Arabic sentiment analysis a very relevant one, albeit a challenge for classification systems. So, this kind of analysis is innovative not only because of the nature

[1] http://www.ArabSocialMediaReport.com/

of the Arabic Twitter dataset used, but also because of the outcomes that it seeks to obtain. Generally, different challenges surround the Arabic Twitter sentiment analysis: limitation in length, as each tweet can contain a maximum of 140 characters [25]. Also, a tweet can contain a significant amount of information, such as embedded links to other websites, hashtags, elongations, abbreviations, emoticons, and unintentional misspellings [20] [21]. Furthermore, many tweets are written in a sarcastic way or with unclear polarity, which makes their meaning very difficult to grasp. Arabic speakers use Modern Standard Arabic (MSA) and Arabic dialect when writing tweets (e.g. "وش" -What, "يسوي" -Do).

In our experiments, we explore the semantic approach in Arabic sentiment analysis to automatically classify Arabic tweets into positive, negative and neutral classes using available Arabic sentiment lexicons, and the effect of text preprocessing on the classifications' accuracy. The main contributions of this paper are as follows:

- Annotate Arabic twitter dataset manually.
- Propose a mechanism for preprocessing Arabic tweets which includes: the remove of all (stop words, proper nouns, punctuation, and white space), spell correction, (ISRI and Light) stemming and POS tagging.
- Translate the annotated Arabic twitter dataset to English by using Google translate.
- Propose our sentiment analysis system that includes a proposed technique which uses both SentiWordNet lexicons: Arabic and English.

The structure of the paper is as follows: Section (2) summarizes the works that are related to the idea of the present study. Section (3) illustrates the methodology followed to develop the work presented in this paper, described with illustrative examples. Section (4) presents the proposed methodology of the Arabic sentiment classification. Section (5) presents an overview of the main results. Section (6) introduces our conclusions and further work.

2. Related Work

Recently, sentiment analysis has become a flourishing field of text mining and natural language processing (NLP). Many studies have been done on sentiment analysis in different types of social media, where some of them focused on Twitter as an extremely popular online microblogging service [10] [27] [28] [5]. There are two main approaches commonly used to determine the sentiment of text, which are the Semantic approach and the Machine Learning approach. Denecke [12] and Hamouda and Rohaim [18] experimented with both approaches. There have been multiple studies using the semantic approach [17] [7] [18] [19].

So, the semantic approach is an unsupervised technique that deals with lexicons of sentiment words such as Bing Liu, MPQA, and SentiWordNet to determine the sentiment orientation of the sentence or document [5] by extracting all sentiment words from the sentence, then summating their polarities to determine if the sentence has an overall positive, negative, or neutral sentiment [15].

2.1 Arabic Sentiment Analysis

Twitter Arabic sentiment analysis is a complex task. Theoretically, the complexity here is that sentiment analysis approaches have been applied to many languages, but it is invalid for applying directly to the Arabic language because the Arabic text needs much manipulation and preprocessing before applying any of these approaches. In [22] [8] [24] they attempted to overcome the Arabic morphology challenge in sentiment analysis by applying different methods that improved Arabic sentiment classification performance. Numerous types of sentiment analysis methods designed in English have been investigated; however, there are a fair number of studies dealing with the Arabic language as well [24] [20] [8] [6].

Here we review the sentiment analysis of Arabic studies that are most closely related to ours, where they tried to show that the Arabic sentiment lexicons have a significant impact on classification accuracy and performance.

Mourad and Darwish [24] built an Arabic Subjectivity and Sentiment classifier by employing four lexicons: the MPQA, the ArabSenti lexicons, and two other lexicons (English-MSA and English-Dialect phrase). For classification, they used the Naive Bayes and SVM classifiers. Additionally, they randomly sampled 2,300 tweets with manual annotation. Also, they applied six features: tokenization, stemmer, negation, n-gram, emoticons, and POS tag. Refaee and Rieser [20] evaluated sentiment analysis on Arabic tweets by combining two approaches: the lexicon based, and machine learning. They built a new lexicon which contains a subjectivity lexicon and emoticon. They also used 1580 Arabic tweets which are automatically translated using Google Translator, then labeled manually to a positive and negative emotion.

In another way, Salameh et al. [6] experimented with the use of Arabic to English translation system and determined the impact of translation on sentiment analysis. They developed the NRC-Canada English sentiment analysis system to deal with Arabic text, by using multiple sentiment lexicons. However, the dataset they used contained 3200 Levantine dialectal tweets. Finally, they showed that automatic sentiment analysis of translated text can lead to results that are close to that obtained by Arabic sentiment analysis systems.

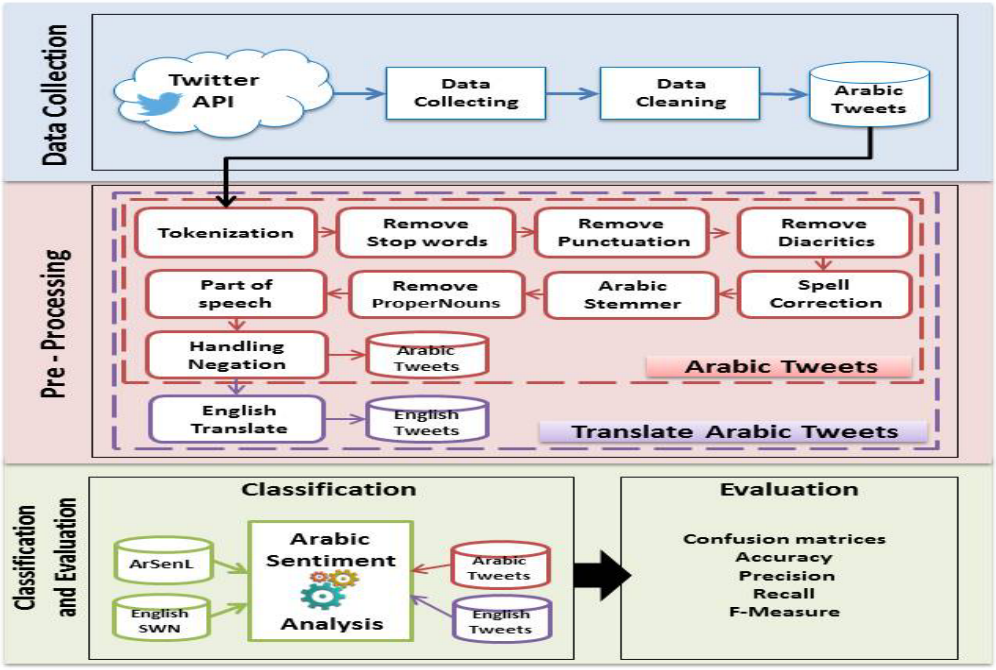

Figure 1. Proposed Arabic sentiment analysis methodology using semantic approach.

As we have seen, most of these researchers used machine learning as a main approach to classify the sentiment of the tweets, which they also labeled to the dataset manually. So, few systems implemented an automatic sentiment analysis of Arabic text by relying only on the semantic approach. To fill this gap, this research proposed a system relying on the Arabic SeniWordNet lexicon to determine the sentiment of Arabic tweets automatically by using a certain number of Arabic linguistic features.

3. Methodology

In the following section, we present the proposed methodology of the Arabic sentiment analysis which includes three main phases (data collection, pre-processing, and sentiment classification and evaluation). Also we describe each phase and the techniques that we used separately.

3.1 Data Collection

Data collection is the first step of our methodology, where we used the Twitter Application Programming Interface (API) with Tweepy[2] library in python which allow access to the Twitter API. A query function was used to request twitter content in the Arabic language as well as a predetermined list of Arabic keywords which covered different topics.

However, we collected the tweets at two times: first, during July 1 to 15, 2015 with 12,384 tweets retrieved, and the second time during September 9 to 12, 2015, with around 2,600 tweets retrieved. The total number of collected tweets was 14,984 tweets, all the tweets that we gathered came with the Unicode format, so we decoded all of them to be meaningful. Then we cleaned them from (#hashtag sign, URLs, @usernames, number, symbols, non-Arabic letters, re-tweets). After the tweets were cleaned, we removed all inapplicable tweets and those containing more than two dialect terms. The total number of tweets was reduced to 3,200.

The final step in data collection was annotating tweets. To annotate the Arabic tweets, we asked for support from two native Arabic annotators, one with a religious background and another in computer science. Moreover, we annotated all 3,200 tweets but in our experiments we used only 2,000 tweets because the remaining 1,200 tweets caused some confusion after we used them. The final annotated tweets comprising our dataset are 634 positive, 675 negative and 691 neutral tweets where the total number of tokens is 26,349.

In addition, we faced multiple issues when determining the class of the tweets: before annotating any tweet, we had to consider the context, the topic of the tweet. In other words, the context of the tweet may flip the sentiment of the tweet. Another issue is that Arabic tweets contain a lot of Islamic supplication, e.g. "Dua'a" which leads to confusion about whether the appropriate class would be positive or neutral. In such cases, we often classified the tweet as positive. On the other hand, while some tweets had a number of positive words, we classified them as neutral because they seemed like an exhortation or a quotation from the Qur'an.

3.2 Proposed Pre-Processing

In the pre-processing phase, we examined a wide variety of Arabic linguistic techniques, which we implemented using Python programming language. Before working on the tweet, we first tokenized it into lists of words, numbers, and symbols which are

[2] http://www.tweepy.org/

called tokens by using NLTK tokenization. Then we removed stop words such as prepositions, articles, and pro-nouns, e.g. (في, هذا, من, أنا, etc.). After that, we removed punctuation and blank spaces. In addition, we removed the diacritics from both the tweets dataset and the Arabic sentiment lexicon to reduce confusion when searching for words in the lexicon. Among the social media platforms, Twitter users' posts tend to be the worst in terms of spelling and grammar. In fact, one in 150 English words posted on Twitter is misspelled [2]. To overcome this problem, we used an Arabic tool called Ghalatawi: Arabic Auto-Correct, which is supported in Python. To reduce the neutral results, we applied the proper noun removal feature, because most proper nouns have a neutral score in Arabic SentiWordNet (ArSenL). However, we used a list of 14,000 Arabic proper nouns including most Arabic family names, cities, etc. Also, the pre-processing phase contains the following tasks:

(1) Arabic Stemmer: As we know, the words in the Arabic language are derived from sets of roots that describe a basic idea with added affixes, which change the word's pronunciation [1]. So, in our system we used two main types of stemming processes that apply to the Arabic language: a Root extraction stemmer, such as The Information Science Research Institute (ISRI) Stemmer; the second type is Light stemmer.

Row Tweet:	لون الارض في احضان بحارها التي ترفع كفيها نحو السماء تسأل الله النصر و السلام و النصر يحتضنها و سماما ع صدره المتن لأنها الوطن The color of the ground in the embrace of seas that raise her hands toward the sky ask God's for victory and peace, victory hug like insignia on his chest because she is the homeland
ISRI Stemmer:	لون ارض احضن بحر رفع كفي سمء تسل الل نصر سلم نصر حضن سما ع صدر متن لأن وطن
Light Stemmer:	رض حضن حار رفع ف سماء سال ل نصر سلام نصر حتض ام ع صدر وطن

Figure 2. Arabic stemmer example.

Using Arabic stemmers can be quite problematic, since there are no accurate Arabic stemmer tools we can rely on to use in our experiment and support python program language. Therefore, we carried out some additional steps to increase the level of accuracy. First the system will search for a word without using the stem in ArSenL; if it cannot find the sentiment of the word, then it will stem the word by ISRI stemmer; if a word cannot be stemmed, then it will go to the second approach, the Light stemmer, instead of leaving the word unchanged.

(2) Part of Speech (POS): Arabic has a scientific grammatical system, where an Arabic word can be classified into three major parts of speech: Nominal (Ism - اسم), Verb (Fia'al- فعل) or Particle (Harf - حرف) [16]. We used POS tagging to choose accurate synsets, because the SentiWordNet lexicon relied on POS for classification. There is only one Arabic POS tagging tool supported by Python; it is NLP Sanford POS tagging and it is less accurate than MADAMIRA[3] which, unfortunately, is not supported by Python. The NLP Sanford POS tool reads Arabic text and assigns parts of speech to each word using Penn Treebank Tags. However, before operating this tool, we had to download it on a machine that was equipped with Java.

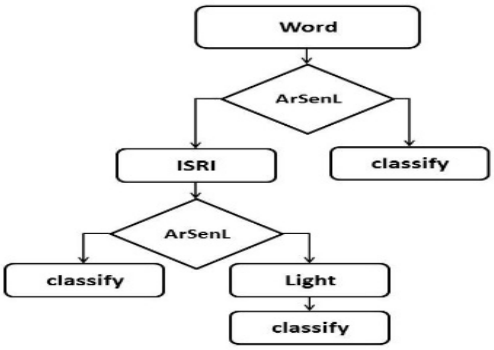

Figure 3. Flow chart of proposed Arabic stemmer technique.

(3) Handling Negation: Negation plays a fundamental role in sentiment analysis by affecting the text polarity. Furthermore, negation handling will be more difficult in the implicit form, where the sentence carries a negative sentiment without the use of negative words, such as sarcastic sentences, e.g. ("This is the first and last time ", "هذي أول و أخر مرة"). For the purpose of the study, we collected a list of the most frequently used negation words in the Arabic language; when the classifier found such negation words, it would give the word in the list a score of (negative = 1). The goal of this method of handling negations is to increase the total of the negative score in a sentence.

3.3 Arabic Sentiment Lexicon

For Arabic Sentiment analysis we used the Arabic SentiWordNet lexicon (ArSenL), created by Badaro et al. [8], and which is similar to the English SentiWordNet. In addition, the ArSenL lexicon contains 28,780 lemmas and 157,969 synsets [14]; each lemma in the lexicon has three scores associated with three sentiment labels (positive, negative, and objective).

4. Sentiment Classification

There are different ways to calculate the sentiment of a text when using SentiWordNet as a sentiment lexicon. SentiWordNet lexicon assigns different sentiment weights to different words. However, the classification begins with a search for words in SWN with their corresponding synsets. Each word in SWN contains numbers of synset, and each synset contains three types of scores: positive, negative, and neutral.

[3] http://nlp.ldeo.columbia.edu/madamira/

For classification, first calculate the arithmetic average of the positive, negative and neutral scores of each word synsets of the tweet by summing up the different scores of each synset class and dividing by the total number of synsets, see equation (1):

$$\begin{aligned} \text{Score pos (W)} &= \frac{1}{n}\sum_{i=1}^{n} \text{score pos (Si)} \\ \text{Score neg (W)} &= \frac{1}{n}\sum_{i=1}^{n} \text{score neg (Si)} \\ \text{Score neu (W)} &= \frac{1}{n}\sum_{i=1}^{n} \text{score neu (Si)} \end{aligned} \quad (1)$$

Where S is the score of synset i, for word, and n is the number of synsets of word, where W is a word that belongs to a set of words.
The second step in classification is determined by the sentiment scores of the tweet by averaging each word score in T, where it represents a single tweet from the set of tweets; see equation (2):

$$\begin{aligned} \text{Total Score pos (T)} &= \frac{1}{n}\sum_{i=1}^{n} \text{score pos (Wi)} \\ \text{Total Score neg (T)} &= \frac{1}{n}\sum_{i=1}^{n} \text{score neg (Wi)} \\ \text{Total Score neu (T)} &= \frac{1}{n}\sum_{i=1}^{n} \text{score neu (Wi)} \end{aligned} \quad (2)$$

4.1 In Formula (1) [13],[23]

The class of the tweet corresponds to the greatest positive, negative, and neutral score:

$$\text{Sentiment}=\begin{cases} \text{Positive tweet if } \text{TotalScore}_{pos}(T) > \text{TotalScore}_{neg}(T) \\ \text{Negative tweet if } \text{TotalScore}_{pos}(T) < \text{TotalScore}_{neg}(T) \\ \text{Neutral tweet if } \text{TotalScore}_{pos}(T) = \text{TotalScore}_{neg}(T) \end{cases}$$

4.2 In Formula (2) [17],[19],[26],[9],[22],[11]

The difference of the total positive and negative scores of the tweet represents its overall sentiment score. In addition, the overall score will be in the range (-1, 1) where a value greater than zero denotes a positive sentiment, and a value less than zero denotes a negative sentiment. Otherwise, the tweet is neutral.

Total Score= Total Score pos (T) – Total Score neg (T)

$$\text{Sentiment}=\begin{cases} \text{Positive} & \text{if Total Score} > 0 \\ \text{Negative} & \text{if Total Score} < 0 \\ \text{Neutral} & \text{if TotalScore} = 0 \end{cases}$$

4.3 Proposed Technique to use Arabic SWN Lexicon in Arabic tweet

To improve the efficiency of using SentiWordNet in sentiment classification for Arabic tweets, we propose a technique that consists of two phases:

(1) SentiWordNet Interpretation Phase

As previously mentioned, each word in SentiWordNet lexicon can have multiple synsets. Therefore, to assign the positive, negative and neutral scores to each word in the text, we need to be performing word synset disambiguation. Initially we removed the unnecessary words from each tweet. Then, we had to import the Arabic Sentiment lexicon (ArSenL) as an MS excel sheet; after that, we looked up each word to check if it belonged to the list of Arabic negation words; if true, the sentiment score was assigned as negative (Scoreneg= 1; Scorepos= 0; Scoreneu = 0). If not, we searched it in ArSenL; if we found it there, we would take all possible synsets scores that are relevant with the POS of the word. If we did not find it, the word would be stemmed by ISRI and then we searched for it again in the ArSenL; also if we did not find it in the lexicon, we would stem the word by Light Stemmer as a second option to find the word in the lexicon. Finally, if the word was not found there either, we would assign a zero value to all scores.

Algorithm 1 : Word Score Summation

```
if WordAveragPositive >= 0 then
    TotalWord= WordAveragPositive - WordAveragNegative
    if TotalWord > 0 then
        TotalPositiveScor+= WordAveragPositive
    end
    else if TotalWord < 0 then
        TotalNegativeScor+= WordAveragNegative
    end
    else
        TotalNeutralScor+= WordAveragNeutral
    end
end
if WordAveragPositive == 0 then
    TotalWord= WordAveragNeutral - WordAveragNegative
    if TotalWord >= 0 then
        TotalNeutralScor+= WordAveragNeutral
    end
    else if TotalWord < 0 then
        TotalNegativeScor+= WordAveragNegative
    end
end
```

Figure 4. Algorithm1.

(2) Sentiment Calculations Phase

In this phase, we apply two calculation methods on tweets to classify them as positive, negative or neutral using word scores obtained from the first phase.

Word score summation method: In this method, we applied Algorithm (1) where summation is the positive, negative and neutral scores for each word in a tweet then calculated the average of each category's scores in each word using Equations (2). Then, to determine the sentiment of the word we followed Formula 2, where we took only the result score in the second step and ignored the other score. To classify each word, we had to calculate the total result of each class (positive, negative and neutral) score. Before applying Algorithm (2) we only considered the result of each word to calculate the total result of the tweet and ignored the other scores.

Tweet score summation method: In this method, we summarize all the positive, negative and neutral scores obtained from the first step. We applied Algorithm (2) to solve the neutral classification issue; we implemented a condition that if the total positive score equals or is more than 0.25, then we performed Formula 2; if not, we implemented another condition, which is: if the difference of the total negative score and the total neutral score is less than zero if the total negative score is more 0.25, then the sentiment of the

tweet is negative. Since all words in the SentiWordNet lexicon have three values (positive, negative and neutral), most of the words have low positive and negative scores and high neutral scores. So, we applied these thresholds to reduce the neutral results.

4.4 Evaluation Measures

In order to decide whether the classifiers were accurately capturing a pattern, we evaluated the model. Furthermore, confusion matrices, precision, recall, F-measure and accuracy methods were used for evaluating the proposed classifiers performance and comparing them with each other.

Algorithm 2 : Tweet Score Summation

```
if TotalPositiveScore >= 0.25 then
    TotalTweet=TotalPositiveScore-TotalNegativeScor
    if TotalTweet > 0 then
        Print Tweet is Positive
    end
    else if TotalTweet < 0 then
        Print Tweet is Negative
    end
    else
        Print Tweet is Neutral
    end
end
else
    TotalTweet=TotalNeutralScor-TotalNegativeScor
    if TotalTweet < 0 Or TotalNegativeScor >= 0.25 then
        Print Tweet is Negative
    end
    else if TotalTweet < 0 then
        Print Tweet is Neutral
    end
end
```

Figure 5. Algorithm 2.

5. Results And Discussion

In this section, we evaluate our semantic approach in a Twitter Arabic sentiment analysis by two experiments performed on Arabic tweets: In the first experiment, we used the Arabic Sentiment Lexicon (ArSneL) and in the second, we used the English SentiWordNet. In addition, we presented the results of different experiments by observing the performance of each classifier to obtain the best one for Arabic sentiment analysis. Also, we measured the effectiveness of both sentiment lexicons: Arabic and English.

5.1 Experiment 1

In this experiment, we studied the performance of using the Arabic Sentiment Lexicon (ArSneL) classifier in an Arabic sentiment analysis of Twitter. Also, we tested the effect of preprocessing on the ArSneL classifiers performance, as well as the effect of using Arabic stemming tools, the Arabic NLP Stanford POS tagging tool and negation on the performance. We divided this experiment into four classifiers.

In classifier (1), we used an unlabeled Arabic tweet dataset that consists of 2,000 tweets, which we collected before the first experiment and which were cleaned of noisy Twitter symbols and duplicated tweets. We applied both formulas (1 and 2) and used ArSenL on the unlabeled Arabic tweets dataset that consists of 2,000 tweets, and which was cleaned of noisy Twitter symbols and duplicated tweets. In addition, classifier (2) was applied to the same dataset in classifier (1) but using the proposed technique. In classifier (3), we applied different preprocessing steps to this dataset, such as tokenization, spelling correction and the removal of all punctuation, Arabic stop words, tashkeel, and proper nouns, but there was little impact on the results of the classification. According to the structure of SentiWordNet, part of speech (POS) is considered an important attribute. Therefore, we used POS tagging as a feature in the classifier (4) by using the Arabic NLP Stanford POS Tagging tool. We also used ISRI and Light stemming, as well as handling negation. We applied these features because they may support and accelerate the process of extracting words from the ArSneL lexicon. After extracting scores for each word, we applied our proposed technique for the sentiment classification of Arabic tweets into one of three classes: positive, negative or neutral. All these steps were implemented using Python.

For the evaluation of this experiment, we compared the results of classifying the labeled dataset that we used in the previous experiments by four performance measures: confusion matrices, precision, recall, F-measure and accuracy.

In the first experiment, as shown in Table 2, we applied Formula (1) and (2) and used ArSenL to achieve an accuracy of 41 %, with high precision in the positive class of 72.91 %, but less than reasonable in the neutral class at 2.18 %. Also, in the second classifier we classified the same dataset but used the proposed technique to reduce the neutral result, where the performance accuracy improved, at 42.75 % with the logical distribution of the precision, recall and F–measure percentages.

With regards to improving the performance of the second classifier, we applied different preprocessing steps to the third classifier. However, there was very little improvement in the result of 0.40 %, but, on the other hand, the process was accelerated. Some issues led to this result, which we will explain in detail:We aimed to remove the stop words from the list because some words may cause confusion in classification. For example, ("الذي" - Who) and ("أنا" - I) are stop words that have neutral weight equaling 1 in Arabic and do not add any sentiment to the tweet, so we kept them on the stop words list, but all negation words were removed from the list such as ("لا" and "لم" - No).

Table 1. The synsets of word"من" in ArSneL lexicon.

POS	Positive	Negative	Objective	English Equivalent
V	0.125	0	0.875	yield, grant, concede, allot, accord, award
V	0	0	1	grant, deed_over, confer, bestow, yield, concede, cede, give
V	0.50	0.25	0.25	lend, impart, contribute, bring, bestow, add
N	0.125	0	0.875	favour, favor
N	0	0	1	party_favour, party_favor, favour, favor, thanksgiving, grace, blessing, state_of_grace, saving_grace
N	0.778	0.222	0	grace, goodwill, good_will
N	0.50	0	0.50	grace_of_god, grace, free_grace, seemliness, favour, favor

Table 2. The results of Arabic tweets sentiment analysis classifier.

Arabic Sentiment Lexicon		Confusion matrices			Results			
		Positive	Negative	Neutral	Precision	Recall	F–measure	Accuracy
Formula (1-2)	Positive	463	323	426	72.91 %	31.48 %	43.97 %	41.00 %
	Negative	136	342	248	50.59 %	28.79 %	36.70 %	
	Neutral	36	11	15	2.18 %	2.88 %	2.48 %	
Proposed Technique	Positive	227	111	187	35.75 %	23.55 %	28.39 %	42.75 %
	Negative	108	247	121	36.54 %	25.65 %	30.14 %	
	Neutral	300	318	381	55.30 %	31.28 %	39.96 %	
Preprocessing	Positive	381	269	302	60.00 %	30.14 %	40.13 %	43.15 %
	Negative	74	228	133	33.73 %	24.86 %	28.63 %	
	Neutral	180	179	254	36.87 %	26.57 %	30.88 %	
POS	Positive	376	233	310	59.21 %	29.79 %	39.64 %	42.75 %
	Negative	64	196	96	28.99 %	22.76 %	25.50 %	
	Neutral	195	247	283	41.07 %	27.69 %	33.08 %	

Table 3. The results of translated Arabic tweets sentiment analysis classifier.

English SentiWordNet		Confusion matrices			Results			
		Positive	Negative	Neutral	Precision	Recall	F–measure	Accuracy
Without Preprocessing	Positive	445	254	371	70.08 %	23.07 %	34.71 %	45.60 %
	Negative	65	257	108	38.02 %	27.75 %	32.08 %	
	Neutral	125	165	210	30.48 %	25.64 %	27.85 %	
Preprocessing	Positive	312	204	217	49.13 %	27.88 %	35.58 %	43.50 %
	Negative	51	160	74	23.67 %	20.67 %	22.07 %	
	Neutral	272	312	398	57.76 %	32.17 %	41.33 %	
POS	Positive	448	275	363	70.55 %	32.18 %	44.20 %	43.45 %
	Negative	48	178	83	26.33 %	21.95 %	23.94 %	
	Neutral	139	223	243	35.27 %	26.19 %	30.06 %	

There are other words we could not remove such as ("من"- Mn), which in most Arabic tweets means from or who. On the other hand, the word ("من"- Mn) could mean grant or favor or dead_over, but when it is written with tashkeel "مَنّ", it is different from "مَنْ", where the former means grant and the latter means who. In Table 1, the ArSenL lexicon has different synsets for the word "من"with the total scores being positive = 3.528, negative = 0.597, neutral = 5.875. However, before removing any stop words, we must remove the tashkeel from the words to avoid these problems. Another issue is misspellings. While a small percentage of tweets are correct and do not include any misspelled words, most mistakes are made in vowel letters; for example "نتهمة" should be "نتهمه" with (ha' alddamir - هاء الضمير) not (altta' almarbuta- التاء المربوطة), the other word "إستعمارى", should not include hamzah "استعماري", in the word "اوربا" the "و" should be included to be "اوروبا", and the word "اليونانين" should be written as plural (Greek), but with the "ي" letter included at the end to be " اليونانيين ". However, all these words and those like them must be preprocessed regarding spelling correction and stemming before classification.

Means	Tweet
From	يارب توفق الحبه و يرجع من طهران بثلاث نقاط و صداره
	Oh God, help my lover (Alhilal FC) and back from Tehran with three points and be the best
Who	من لم يتقن فن الصمت لن يتقن فن الكلام
	Those who have not mastered in silent skill will not mastering in speech skill

Figure 6. Word "من" examples.

In the POS classifier, we added the POS tagging feature to our Arabic sentiment classification. We expected POS would play a major role in sentiment analysis because it is an important attribute in sentiment lexicons. However when using POS tagging, there have not been any noticeable improvements in the performance results of the Arabic SWN classifier at almost 42.75%, aside from accelerating the search process.

On the other hand, we also applied the stemming feature in the POS classifier, including both the Arabic stemmer methods ISRI and Light sequentially; as we explained, the previous ISRI method relies on extracting the root of the word by looking it up in a root dictionary. Therefore, we applied the first step, and if the root of the word was not found, the classifier would go to the next step, Light stemming. This technique allowed us to take advantage of both methods, which may provide a greater possibility of finding a word in the ArSenL lexicon. ISRI Arabic stemming is generally a good method, but it requires updating and a better review of roots because there are many mistakes; for example, if we stemmed the word "بارضنا" meaning our land, it becomes "برض" and this is incorrect. However, if we apply our technique it will not solve the problem because the word "برض" has a score in the ArSenL lexicon.

5.2 Experiment 2

In this experiment, we followed the same steps as in the first experiment, but we used English tweets and English SentiWordNet. Initially, we translated all the Arabic tweets dataset to English using Google Translator API. The total number of tweets we translated was 2,000 Arabic tweets; however we translated twice: the first time without preprocessing and the second after preprocessing. In Table 3, it is clear that when we classified the translated tweets without any preprocessing, the classifier works well, producing fewer classification errors with an accuracy of 45.60%, but in the second classifier after preprocessing the accuracy was 43.50%. In the third classifier, we applied the English POS tagging feature on the translated tweets; after preprocessing we noticed no improvement in accuracy as a whole, but the Positive precision clearly improved with 21.42% from the first classifier.

Furthermore, this experiment revealed a number of issues that led to an incorrect translation of the Arabic tweets, and in addition, the classifier was not able to manage this:

One of these issues was that some words, such as "كوباني" were translated to be "KobânÎ", contained non-English letters such as (â, è, î, ü), so English SWN does not contain words with these letters.

Also there were many misspelled words in the Arabic tweets, as previously mentioned, which is a problem that continued to be one reason for incorrect translations, for example in this tweet " اقسم بالله ان اهل اليمن كللهم شرفاء ", the word "كللهم", which mean "All" is incorrect, so when we translate, it becomes "swear god people kllahm honest". Another issue was that some derivative words could not be translated by Google (see Figure 7).

Ar	أعشق قراءة تغريداتها كل يوم حتى و أن لم أتابعها فكل تغريدة تكبتها لها ذكرى بداخل
En	Adore reading ngredatha otabaha every tweet tkptha memory inside

Figure 7. Google translate examples.

With regard to this issue, the tweet in Figure 7 contains the word "تغريداتها"-"her tweets", derived from the word "تغريدة" - "tweets", which Google failed to translate, and also there is the misspelled word, "تكبتها"; the correct word should be "تكتبها" - "Writes it". All these words were neglected by the classifier, which contributed to the low result. Thus, the correct translation for this tweet if there were no misspellings is "I love to read her tweets every day, even if I do not follow her. Any tweet she writes has a memory inside me".

Another issue that appeared in this experiment was with homonyms, which are a group of words that share the same spelling but that have different meanings such as "like" (similar or comparable) and "like" (agreeable or enjoyable). Furthermore, the classifier could not distinguish between the meanings of words, so we had to overcome this issue by translating each Arabic word individually. For example, the translation of the sentence "أنا أحب التفاح" becomes "I like apple"; however, the same sentence translated manually is, "I love apple".

Arabic sentence	أنا أحب التفاح		
Translate sentence	I like apple		
Separate words of sentence	التفاح	أحب	أنا
Translate each word	Apple	Love	I

Figure 8. Homonym example.

By using the first method, the sentiment classifier classified words "like" to neutral with total scores of scorepos=2, scoreneg=0 and scoreneu=3; however, when classifying the word "love", it was positive with total scores of scorepos=2.5, scoreneg=0.125 and scoreneu=1.375.

5.3 Comparison

Table 4. Accuracy of sentiment analysis systems on BBN datasets.

System	Lexicon	Accuracy
Our system	Arabic SentiWordNet	60,25%
SMK system [6]	Arabic Hashtag Lexicon(dialectal)	63.89%

We tested our system on the existing BBN Blog Posts Sentiment dataset [4]. Table 4 shows the results of ten-fold cross-validation experiments on the dataset, where the results shown for systems have to identify one of three classes: positive, negative, or neutral. The accuracy of our system is convergent to previously published results by Salameh et al. system [6], where they obtained an accuracy of 63.89%.

6. Conclusion And Future Work

In this paper, we explored a new model of Arabic tweets sentiment analysis using a semantic approach, and the effect of different techniques of text preprocessing on the classifications accuracy. The main contribution of this paper is the implementation of a novel Arabic tweets sentiment analysis system by using both Arabic and English sentiment lexicons. With regard to the results, it is clear there are no noticeable effects on performance when applying preprocessing and the POS tagging feature. This is because there are a lot of misspellings and dialect words in Arabic tweets. Moreover, most of the good text preprocessing tools do not support the Python programming language. Furthermore, the stemmer tools produced an incorrect root or a root without meaning. Hence, we conclude that there are two tools that must be improved to support and improve any Arabic natural learning process system. The first is the Arabic spelling correction tools, which need to be developed and more words added. The second is the Arabic stemmer tools that need to include more roots in order to support the searching process.

References

[1] S. M. Oraby, Y. El-Sonbaty, and M. A. El-Nasr, Exploring the Effects of Word Roots for Arabic Sentiment Analysis, in *Proceedings of the International Joint Conference on Natural Language Processing.* Nagoya, Japan, pp.471-479, 2013.

[2] J. O'Mahony, Twitter users 'can't spell' Telegraph, http://www.telegraph.co.uk/technology/twitter/10086819/Twitter-users-cant-spell.html, December 2015.

[3] G. Wang, J. Sun, J. Ma, K. Xu, and J. Gu, Sentiment classification: The contribution of ensemble learning, in *Decision support systems*, pp.77-93, 2014.

[4] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, Sentiment analysis of twitter data, in *Proceedings of the Workshop on Languages in Social Media*, pp. 30-38, 2011.

[5] S. Asur, and B. Huberman, Predicting the future with social media, in *Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pp. 492-499, 2010.

[6] M. Salameh, S. M. Mohammad, and S. Kiritchenko, Sentiment After Translation: A Case-Study on Arabic Social Media Posts, in *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL-2015)*, 2015.

[7] Y. Yu, and X. Wang, World Cup 2014 in the Twitter World: A big data analysis of sentiments in US sports fans' tweets, *Computers in Human Behavior*, pp. 392-400, 2015.

[8] A. R. Hedar, and M. M. Doss, Mining Social Networks' Arabic Slang Comments, in *Proceedings of IADIS European Conference on Data Mining (ECDM'13)*. Prague, Czech Republic, 2013.

[9] K. Denecke, Using sentiwordnet for multilingual sentiment analysis, in *Data Engineering Workshop, ICDEW 2008, IEEE 24th International Conference*, pp. 507-512, 2008.

[10] M. Anjaria, and R. M. R. Guddeti, A novel sentiment analysis of social networks using supervised learning, *Social Network Analysis and Mining*, vol.4, no.1, pp. 1-15, 2014.

[11] A. Montejo-Ráez, E. Martínez-Cámara, M. T. Martín-Valdivia, and L. A. Ureña-López, Ranked wordnet graph for sentiment polarity classification in twitter, *Computer Speech & Language*, vol.28, no.1, pp. 93-107, 2014.

[12] K. Denecke, Are SentiWordNet scores suited for multi-domain sentiment classification?, in *Digital Information Management, ICDIM 2009, Fourth International Conference*, pp. 1-6, 2009.

[13] M. Guerini, L. Gatti, and M. Turchi, Sentiment analysis: How to derive prior polarities from SentiWordNet, The arXiv, pp. 1309-5843, 2013.

[4] http://saifmohammad.com/WebPages/ArabicSA.html

[14] G. Badaro, R. Baly, H. Hajj, N. Habash, and W. El-Hajj, A large scale Arabic sentiment lexicon for Arabic opinion mining, *ANLP 2014*, 2014.

[15] A. Shoukry, and A. Rafea, Sentence-level Arabic sentiment analysis, in *Proceedings in International Conference of the Collaboration Technologies and Systems (CTS)*, pp. 546-550, 2012.

[16] S. Khoja, APT: Arabic part-of-speech tagger, in *Proceedings of the Student Workshop at NAACL*, pp. 20-25, 2001.

[17] E. Martınez-Cámara, S. M. Jiménez-Zafra, M. T. Martın-Valdivia, and L. A. Urena-López, SINAI: Voting System for Twitter Sentiment Analysis, *SemEval 2014*, 572, 2014.

[18] A. Hamouda, and M. Rohaim, Reviews classification using sentiwordnet lexicon, In *World Congress on Computer Science and Information Technology*, 2011.

[19] F. H. Khan, S. Bashir, and U. Qamar, TOM: Twitter opinion mining framework using hybrid classification scheme, *Decision Support Systems*, pp. 245-257, 2014.

[20] E. Refaee, and V. Rieser, Evaluating Distant Supervision for Subjectivity and Sentiment Analysis on Arabic Twitter Feeds, *ANLP 2014*, 2014.

[21] N. El-Makky, K. Nagi, A. El-Ebshihy, E. Apady, O. Hafez, S. Mostafa, and S. Ibrahim, Sentiment Analysis of Colloquial Arabic Tweets, 2015.

[22] M. Abdalkader, Sentiment Analysis of Egyptian Arabic in Social Media, 2014.

[23] R. Ortega, A. Fonseca, M. Mendoza, and Y. Guti'errez, SSA-UO: unsupervised Twitter sentiment analysis, in *Second Joint Conference on Lexicon and Computational Semantic*s, vol.2, pp. 501-507, 2013.

[24] A. Mourad, and K. Darwish, Subjectivity and sentiment analysis of modern standard Arabic and Arabic microblogs, in *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity*, Sentiment and Social Media Analysis, pp. 55-64, June 2013.

[25] A. S. Aruna, and P. K. Wilson, Sentiment Analysis using Linguistic Structures-(Adv-Adj-Noun), in *International Journal of Advanced Research in Computer Science*, vol.5, no.1, 2014.

[26] R. Pandarachalil, S. Sendhilkumar, and G. S. Mahalakshmi, Twitter Sentiment Analysis for Large-Scale Data: An Unsupervised Approach, *Cognitive Computation*, pp. 1-9, 2015.

[27] A. Pak, and P. Paroubek, Twitter as a Corpus for Sentiment Analysis and Opinion Mining, in *LREC*, vol.10, pp. 1320-1326, 2010.

[28] X. Zhu, S. Kiritchenko, and S. M. Mohammad, Nrc-canada-2014: Recent improvements in the sentiment analysis of tweets, *SemEval 2014*, 2014.

Lamia Ali Al-Horaibi received her M.S. degree in Information Systems from Al-Imam Muhammad Ibn Saud Islamic University (ImamU), Saudi Arabia, in 2016. She is working now as system analyst and developer in Princess Nourah bint Abdulrahman University, Saudi Arabia.

Muhammad Badruddin Khan He received his PhD degree from Tokyo Institute of Technology, Japan. He is currently an assistant professor in the Department of Information Systems, College of Computer and Information Sciences, Al-Imam Muhammad ibn Saud Islamic University.