

Data and Text Mining Techniques for Classifying Arabic Tweet Polarity

Belgacem Brahim¹, Mohamed Touahria², Abdelkamel Tari³

¹ Department of Computer Science
University of A. Mira, Bejaia, 06000, Algeria

² Department of Computer Science
University of Sétif, Sétif, 19000, Algeria

³ Department of Computer Science
University of A. Mira, Bejaia, 06000, Algeria

belkacem.brahimi@yahoo.fr, touahria_momo@yahoo.fr, Tarikamel59@gmail.com



*Journal of Digital
Information Management*

ABSTRACT: *Sentiment analysis is a new task related to text mining that extracts opinions from textual data and classifies them into positive, negative or neutral. The goal of this paper is to determine the effect of applying stemming and n-gram techniques for Arabic texts (tweets) on sentiment classification. This study also aims at investigating the impact of feature selection on the performance of the classifier. For this reason, three classifiers Support Vector Machines (SVM), Naïve Bayes, (NB), and K-nearest neighbor (KNN) are used. The obtained results showed that the best results of performance are obtained when applying a hybrid representation which includes tokens with character 3-grams. The experiment results also revealed that the use of feature selection technique improves significantly the accuracy of the three classifiers for the task of opinion classification. Regarding The classifiers, SVM outperforms the other classifiers when using all the features, while when selecting the most relevant features by the SVM feature selection technique, SVM and NB provided the best results.*

Subject Categories and Descriptors

H.3.1[Content Analysis And Indexing]: H.3.5[Online Information Services]: Web-Based Services; **I.2.7 [Natural Language Processing]:** Text analysis

General Terms: Text Mining, Natural Language Processing

Keywords: Sentiment Analysis, Text Mining, Knowledge Discovery, Stemming, Feature Selection, Text Annotation, Twitter

Received: 10 September 2015; Revised 9 October 2015; Accepted 18 October 2015

1. Introduction

Nowadays we witness a speedy growth and spread of web resources such as Social networking sites, online review sites, personal blogs, which allow users to express and share their ideas, opinions and judgement about different issues. The task of collecting and analyzing these opinions and comments is very important in several real situations. For example, a customer wants to know the opinions of the users when he decides to purchase a product or a service. A company desires to know the customer opinion in order to adapt and improve the quality of its product. In the politic domain, a party is interested in predicting the orientation and the trends of voters.

Text classification is one of the most important tasks in text mining that automatically assigns text documents to one or more predefined categories based on content and linguistic features [1]. This task is very useful for several needs, for example, organizing the huge amount of documents into their proper classes or folders. A typical text classification framework includes the following steps: preprocessing, feature extraction, feature selection, and classification.

Opinion mining (or sentiment analysis) can be regarded as a special case of text classification that tries to automatically extract and determine the orientation (polarity)

of evaluations , emotions, judgements, which are positive, negative or neutral from unstructured texts.

In the present paper, we study the impact of stemming on opinion classification of Arabic tweets. We try to find the best representation of tweets expressed in Arabic for the task of sentiment classification. In the literature, different studies conducted this comparison for Arabic in text classification [2] [3] and sentiment analysis [4]. However, these comparisons do not include the character n-gram technique. In fact, n-gram of characters model is an important difference between the two tools (Rapidminer¹ and weka²) which is not mentioned in the comparative study of [5].

At our best knowledge, the only detailed work on the impact of the preprocessing step on Arabic sentiment analysis was conducted by [6]. For this end, we performed a comparison between the two well known stemming techniques (light stemming and root stemming) and character n-grams technique for tweets expressed in Arabic. Furthermore, regarding the feature selection problem, we study the effect of selecting the most important features on classification accuracy. To achieve this goal, we employed the three well known classifiers which are Support Vector Machines (SVM), Naïve Bayes (NB), and K-nearest neighbor (KNN).

The remainder of this paper is structured as follows. Section two gives an overview about Arabic language, feature selection and sentiment analysis. In section three, we present recent work related to opinion classification for Arabic. Section four introduces our system proposed for classifying tweets (datasets, classifiers, etc.). In section five; we provide the results of experiments and their interpretations. Finally, a conclusion and future work are given in section six.

2. Background

2.1 Arabic Language

The Arabic Language is the 5th widely used languages in the world. It is spoken by more than 422 million people as a first language and by 250 million as a second language [7]. Arabic alphabet consists of 28 letters. There is no upper or lower case for Arabic letters like English letters. The letters (ي و ا) are vowels. The orientation of writing in Arabic is from right to left [8].

The automatic treatment and analysis of Arabic is a very hard task because it is a highly inflectional and derivational language, and unlike English it has a very complex morphology. The tokenization step in Arabic is not easy because prepositions (حروف الجر) and conjunctions such as (and, for, in Arabic ل و) are attached with the word. The stemming stage is also hard

because the word could have many derivations which change the stem itself, so, it is not easy to distinguish between root letters and affix letters. Stemming is the process for reducing inflected (or sometimes derived) word to its stem, base or root form.

In the Arabic Language, there are two techniques; stemming and light stemming. Stemming reduces words to their stems [9], while light stemming removes common affixes from words without reducing them to their roots. For example, Stemming would reduce the words « الكتاب » (in English the book) to the stem « كتب » (to write), while light stemmer reduces the word « الكتاب » to « كتاب » (a book in English)

The intuitive idea for utilizing light stemming is that many word derivations have different meanings or semantics [10]. However, these word variants are generated from the same root. Thus, root extraction algorithms could affect the meanings of words. Light stemming aims to enhance the classification performance while retaining the words meanings. It removes some defined prefixes and suffixes from the word rather than extracting the original root.

Stemming algorithm proposed by Khoja [9] is one of well known Arabic stemmers. This aggressive algorithm removes the longest suffix and the longest prefix. It then matches the remaining word with verbal and noun patterns, to extract the root [11].

The other technique is n-gram which is language independent method and works well in the case of noisy-text. The n-gram based approaches group together words that contain identical character sub-strings of length n [12]. As a definition, character n-gram is a contiguous sequence of n characters from a given sequence of text.

For example, the tri-grams for the word (الكتاب) are

(الف- لكتب- كت- تاب) while the 4-grams of this word are

(الكتب- لكت- كتاب).

The intuitive idea is that the character structure of the word can be used to find semantically similar words and word variants [13]. This technique is independent of language and unlike stemmers; it does not require any language knowledge [14]. The other advantage of n-gram approach is its tolerance to the misspelled and the transliterated words [15].

2.2 Feature Selection

The goal of feature selection technique is to find the most relevant features for the classification task, and to remove irrelevant, redundant, and noisy data [16]. Feature selection has as a second objective the reduction of the dimensionality of feature space and time processing. Accurate feature selection is very important for the performance of any classifier.

SVM-Based feature selection (SVM) technique uses the

¹<https://rapidminer.com/>

²www.cs.waikato.ac.nz/ml/weka/

coefficients of the normal vector of a linear SVM as attribute weights with respect to the label attribute. The attributes with higher weight are considered more relevant. The SVMs technique uses a hyperplane to separate the positive from negative instances. The linear classifier classifies new examples by testing whether the linear combination of the components of vector x is above or below a given threshold.

2.3 Sentiment Analysis

Sentiment analysis is a new research field that uses text and data mining techniques to extract emotions, opinions, evaluations, and feelings about a given target (subject, person, product, service, etc.) and assigns them to positive or negative class (in some cases the neutral and mix classes are considered). In the literature we can find similar terms related to this area of research: sentiment analysis, opinion mining, sentiment classification, sentiment orientation and subjectivity analysis. This task of analyzing opinions has a wide range of applications in real life such as product reviews, advertising systems, bias detection, public relations, etc.

In opinion mining there are generally three different levels: word (feature) level, sentence level and document level [17]. Working on term (word) level consists of identifying the orientation of each word in the text (positive or negative). The goal of sentence level is to classify each sentence in the document. Finally, the analysis conducted on document level focuses on labeling the entire document as positive or negative.

In addition, in order to extract and analyse sentiments, two methods are commonly employed: lexicon-based (semantic methods) [18] and machine learning (ML) based methods [19]. Other approaches use combined methods [20], [21]. The lexicon-based method consists of using a list of words (sentiment words or dictionary). Each word of the list is assigned with a positive or negative sentiment, e.g., good, bad, etc. On the other hand, machine learning (ML) based methods use classification algorithms to classify a review as positive or negative. In the paper [19], Pang founded that standard machine learning methods were very useful for the task of opinion classification. Several studies also showed that machine learning methods were more suitable for twitter than semantic approaches [22], [23].

3. Literature Review

In this section we present the most important studies that have been conducted in Arabic sentiment analysis. We can say that most of the works on sentiment analysis and opinion mining consider only English language. The reasons for limited studies in the Arabic language are the absence of free lexical resources and the complexity of the automatic analysis of Arabic language.

Concerning the n-grams methods, there were several papers that study the use of n-grams for processing

Arabic text. Mayfield et al. [24] showed that the n-grams technique works well in various languages; they explored the use of character n-grams for Arabic retrieval in TREC-2001 and indicated that n-grams of length 4 were most effective. The paper of [25] investigates the use of the digram (2 characters) and trigram (3 characters) term conflation techniques for the task of Arabic text retrieval.

The work of [21] proposed a combined method for Arabic sentiment classification. The idea behind this proposition is that using only one method for the sentiment classification task is not effective and provides a low performance. The author used three classification methods which are lexicon-based method, maximum entropy and KNN for classifying reviews orientation. The lexicon-based approach was first used to predict the document polarity by utilizing a dictionary of lexicon words (positive, negative). After this, the labeled documents by the lexicon method were used as training examples for the maximum entropy classifier. Finally, the rest of documents was classified by the third algorithm which is KKN. This classifier employed the labeled documents from the two previous methods as training set. The results of this work showed that the accuracy is improved from 60% when using only the lexicon method to 80% by using the hybrid approach which combined the three methods.

The authors of [4] studied the impact of the stemming on the opinion classification problem. They performed experiments on two datasets (the PATB, Part 1 v 4.1) [26] and OCA corpus [27]. The stemmers used in their work are Khoja Arabic Stemmer [9], (ISRI) stemmer [28] and Tashaphyne Light Arabic Stemmer [29]. The results showed that for the PATB corpus, the best stemming approach is Khoja stemmer with word unigram, whereas Tashaphyne stemmer added to word unigram gives the best performance in the OCA movie dataset.

The work of Duwairy [6] studied different aspects that affect the classification performance (text representation, feature selection and the choice of the appropriate algorithm). The authors used two datasets. One is about politic issues which includes 300 reviews. The second corpus utilized in this study is the OCA movie dataset [27]. For the categorization task, three algorithms were employed (SVM, Naïve Bayes, and K-nearest neighbor). The results indicated that the performance of the classification methods was dependent on the preprocessing technique and the corpus utilized.

In the study of Abdul-Mageed et al. [30], a system called SAMAR for Sentiment Analysis is proposed. First, it tries to identify whether the text is objective or subjective. After this, the system determines its polarity. The corpus used was compiled from different genres of social media websites: chat, Twitter, Web forums, etc. The authors used features such as lexical features, POS (part of speech) and morphological ones. SVM light algorithm is used for the classification task and provides an accuracy equals 81.36.

Rushdi-Saleh et al [27] used SVM and NB for classifying movie reviews (250 positive and 250 negative) collected from different web pages. The authors applied different preprocessing tasks (manual spelling correction, stop-words removal, stemming, and N-Gram generation). Their experimentation results showed that using SVM yields the best classification accuracy rate (90%). The authors proposed as future works to use the WordNet database for the Arabic language and English resources such as SentiWordNet in order to improve their corpus. They also suggested the automatic translation of the OCA corpus into English for evaluating and analysing the results.

The work of [31] presented two approaches for sentiment analysis in tweets. The researchers compiled a dataset of 2000 tweets which is freely available for research purpose. The authors compared the two main approaches: lexical and machine learning algorithms. The used lexicon included 3479 words (1262 positive, 2217 negative). For the machine learning approach, the authors employed four classifiers (SVM, NB, KNN and Decision Tree DT).

The result of this work revealed that SVM for classification of a light-stemmed dataset gives the highest accuracy. For further work, the researchers proposed to enrich the lexicon by adding a third class which is the neutral category, and studying the sarcasm problem in Arabic sentiment analysis.

In [32] Shoukry and Refae collected a tweet dataset (500 positives and 500 are negative). Unigram-based and Bigram-based features extraction techniques were applied in this work. For the classifiers, the authors used SVM and NB. Their experiment results indicated that SVM outperformed NB.

Mountassir A et al [33] performed a study on Arabic sentiment analysis. The authors used two collections of documents; one is ACOM (Arabic Corpus for Opinion Mining) collected by the researchers. The second collection employed by the authors is OCA [27]. The objective of this work is to investigate the machine learning classifiers and some settings in Arabic sentiment classification. The result of this paper showed that SVM and NB are competitive, while the performance of KNN depends on the collection of the documents.

The study [34] proposed a system for mining Saudi public comments from e-newspaper. The authors used a naïve Bayes algorithm and they revised the bigram model. They also suggested as a future research to deal with the negation effects problem and the generation of the review summary.

Concerning the feature selection technique, at our best of language, the only detailed study concerning the effect of the feature selection technique on Arabic sentiment analysis was [35]. This work presents an empirical comparison of seven feature selection methods

(Information Gain, Principal Components Analysis, Relief-F, Gini Index, Uncertainty, Chi-squared, and Support Vector Machines), and three classifiers (SVM, Naïve Bayes, and K-nearest neighbor). The experiment results showed that the SVM classifier combined with the SVM-based feature selection method gives the best accuracy.

From this survey, we can conclude that the machine learning methods are the most used when compared with the lexical ones. This is due to the effectiveness of the machine learning classifiers. Furthermore, the lexical (symbolic) approaches are not commonly employed in Arabic sentiment analysis because they require additional (mainly lexical) resources which are not publically available in Arabic. A final remark concerns the collections used in Arabic sentiment (tweets, movie reviews, etc.), which are of modest size compared with English datasets.

4. Proposed Approach

In this section, we present our methodology used for the task of classifying the tweet orientation (our text models, datasets, classifiers used, etc.). We selected the tweets collections because Twitter is the most known micro-blogging application in the web. Tweets are popular and suitable for the sentence level classification. As aforementioned in section 2.3, there are three approaches to study the sentiment classification, machine learning approaches, lexical approaches, and hybrid approaches combining the two previous methods. In the present research, we use machine learning methods that need a labeled corpus for training and testing the classifier.

This research aims to determine what is the best representation for tweets in Arabic (tokens, n-gram characters, root stemmed or light stemmed words), and the effect of feature selection technique on classification performance (decreasing, increasing or keeping the accuracy).

4.1 Text Representation

Most of the studies on sentiment classification focus on the comparison between the different stemmers [4] [31]. The contribution of this paper is to compare the different stemming techniques with the n-gram based techniques in order to study their effects on classification accuracy of short texts (tweets). This paper also investigates the contribution of combining different vector representations on opinion classification.

Before classifying a given text, applying a sequence of operations is needed. This is called the preprocessing task. The objective of this step is to prepare the document to the classification stage. The preprocessing task includes the following operations:

- **Cleaning:** Consists of cleaning the text by removing irrelevant items such as advertisements, figures.
- **Tokenization** which consists of splitting the text into words (tokens) separated by whitespaces or punctuation

characters. The result of this operation is a set of words.

- **Stemming:** As was previously mentioned in section 2.1, there are two different techniques of stemming in the Arabic language: stemming and light stemming.

- **Stop Word Removal:** Stop words are function words like prepositions, conjunctions, articles, etc., which are not useful for the meaning of the text. This list of words is created manually, and it is language-specific.

- **Term Frequency Thresholding:** In this step, we remove words which occur too frequently in a corpus, and eliminate terms that appear rarely. These terms are not discriminative between documents

After applying these operations, the text is transformed to a vector representation. In this model the weight of the word (feature) is calculated with the respect to the document containing that word. There are several weighting schemes (Boolean weighting, Term Frequency (TF) weighting, Inverse Document Frequency (IDF) weighting, and Term Frequency Inverse Document Frequency (TFIDF)).

The Boolean weighting (presence) is the simplest scheme, in this case the weight of the word is 1 if it occurs in the document and 0 otherwise. TF (term frequency) calculates the raw frequency of a term in a document, i.e., the number of times that a term *t* appears in the document.

The Inverse Document Frequency (IDF) reflects how much information the word provides, i.e., whether the word is common or rare across all documents. To calculate IDF, we divide the total number of documents by the number of documents containing the term, and then taking the logarithm of that fraction.

TFIDF is the most used scheme for the task of information retrieval and text classification because it combines the two previous schemes (TF and IDF). Mathematically, TFIDF is the product of the two values of TF and IDF. This scheme is also used for measuring the importance of a word in a text within a collection.

To investigate the impact of text representation on the classifiers performance, we use different models which are: Baseline vectors, the tweet is tokenized, without preprocessing; stemmed vectors (light stemmer); stemmed vectors (Khoja's stemmer), vectors contain only character *n*-grams; vectors that combine tokens with character *n*-grams (*n* = 3, 4). In table 1, the different models are illustrated.

The idea of using character 3-grams and character 4-grams is that studies such as [24] [25] have found that these models were most accurate for Arabic document retrieval. We used the Rapidminer tool which provides both models; character *n*-gram and word *n*-gram.

4.2 Classifiers Used

For the classification task, three supervised learning algorithms were utilized: support vector machine (SVM), naïve Bayes (NB) and *k* nearest neighbor (KNN). We give below brief descriptions of these classifiers.

SVMs are a set of supervised learning methods used for the task of classification, regression analysis, etc. They are well known technique for text classification. This is due to their effectiveness and performance for the task of text categorization [36] and sentiment classification [37]. In fact, SVMs are effective in high dimensional spaces and provide different kernel functions which can be used for the decision function. This algorithm attempts to build a boundary between two classes by finding a hyperplane or set of hyperplanes to separate the two classes in some kernel space. A good separation is reached by the hyperplane that has the greatest distance to the nearest training-data point of any class.

Text representation	Content
Words Model	Baseline, the text contains the tokens (words) without preprocessing.
Light stem Model	Tokens are stemmed using light stemmer
Stem Model	Tokens are stemmed using Khoja's stemmer.
3- grams model	Vector contains only character 3-grams.
Words+ 3- grams model	Tokens plus character 3-grams
4- grams model	Vector contains only character 4-grams.
Words+4- grams model	Tokens plus character 4-grams.

Table 1. Proposed text representations

The *K* nearest neighbor classification algorithm is an instance-based learning method. This classifier is simple and efficient for the task of text classification. KNN needs only two parameters to choose which are the number of neighbors *K* and the distance metric (Euclidian, cosine, etc.).

Finally, the naïve bayes classifier is a well known algorithm used in text classification. This algorithm is a probabilistic model based on the use of Bayes' theorem in the classifier's decision rule. NB assumes that the value of a particular feature is independent of the values of the other features, for this reason it is called naïve. The algorithm estimates the posterior probability that the document belongs to different classes and classifies it to the class that has the highest posterior probability.

In this study SVM was employed with kernel linear because it gives the best results. In addition, KNN was used with *k* = 9 and similarity = cosine. The choice of *k*

= 9 and cosine similarity gave us the highest value of the accuracy. The other reason for using $k = 9$ is to make our results comparable with the work of [31], which used the same parameter value.

4.3 Corpus

The corpus used in the experiments is “2000 tweets” collected by the authors of [31]. This dataset contains 1000 positive tweets and 1000 negative ones which cover various topics such as: politics and arts. Table 2 gives a description of this dataset.

The second dataset utilized is “*BBN tweets*” chosen randomly [38] from the BBN Arabic-dialect/ English parallel text [39]³. The original dataset contains 1200 tweets (positive, negative and neutral). We removed neutral tweets since we study only opinionated tweets (positive or negative). The final collection includes 498 positive tweets and 575 negative ones.

	Positive	Negative
Number of tweets	1000	1000
Total number of words	7189	9769
Average number of words per tweets	7.19	9.97

Table 2. Description of the tweets dataset according to [31]

4.4 Evaluation Metrics

To compare our results with the outcomes of the study [31], we followed the same setting used in their work. For this end, 5-fold cross-validation was used rather than 10 in the data set (2000 tweets). For the second collection which is BBN, we used 10-fold cross-validation since it is the most used setting in the classification task.

In k-fold cross-validation, the dataset is partitioned into k subsets, performing the classification on one subset (the training set), and validating the model on the rest (k-1) subsets (called the validation set or testing set). This operation is repeated k times for every subset. The validation results are averaged over the k iterations.

Moreover, in order to evaluate the classification techniques, we calculate the most widely used performance measures in the classification task which are precision, recall, and accuracy. We can define these measures as follows:

$$\text{Precision} = \# \text{ correct docs found} / \# \text{ docs found} \quad (1)$$

$$\text{Recall} = \# \text{ correct docs found} / \# \text{ correct docs} \quad (2)$$

$$\text{Accuracy} = \# \text{ correct docs found} / \# \text{ total docs classified} \quad (3)$$

³<http://catalog.ldc.upenn.edu/LDC2012T09>

where # is number and docs denotes the documents.

The precision gives us the rate of labeled documents that are correct, while the recall measures the fraction of true labels found by the system. The accuracy calculates the percentage of true classifications.

4.5 Tools used

In the experiment, we used the Rapidminer software that provides all the steps of the text (data) mining process such as preprocessing, visualization of the results, validation. This environment also includes several machine learning algorithms. We utilized this tool for the task of preprocessing and vector generation model. In addition, Arabic root stemmer of Khoja and light stemmer are implemented and integrated in RapidMiner. For the task of text categorisation, the machine learning algorithms (SVM, KNN and NB) evaluated in our experiments were also performed in the Rapidminer platform.

5. Experimental Results and Analysis

The goal of the present set of tests is to compare the performance of the three classifiers (SVM, NB and KNN) for the different representations proposed. Table 3, 4 and 5, provide the experiment results of these algorithms for the models of the tweets dataset.

Text models	Accuracy	Precision		Recall	
		Neg	Pos	Neg	Pos
words only	85.65	89.83	82.26	80.40	90.90
light stem	87.65	87.39	87.92	88	87.30
Khoja’s stem	86.85	85.74	88.03	88.40	85.30
3-grams	88.20	85.77	90.99	91.60	84.80
3-grams + words	89.45	87.61	91.48	91.90	87
4-grams	87.95	90.07	86.04	85.30	90.60
4-grams + words	87.25	92.09	83.41	81.50	93

Table 3. Performance rate of SVM on 2000 tweets dataset
Table 3 provides the performance percentages of the SVM classifier for the different text representations proposed. The best results are mentioned in bold. The results show that the two techniques (stemming, n-gram of characters) outperform the base model (words). Furthermore, the light stemmer gives better results than the root stemmer of Khoja. The experiments also indicate that different n-gram models (3-and 4-grams models without and combined with the words model) outperform the two stemming techniques.

From this table we can also see that the representation which combines tokens with character 3-gram (3-grams + words) yields the best results of the accuracy, the precision of the positive class and the recall of the negative one. The 4-grams+ words representation provides the highest precision of the negative tweets and

the best recall value of the positive ones.

Table 4 shows the performance of the NB classifier with different representations of the tweets. The representation1 (only words) and 5 (trigrams plus words) give the best results of two metrics. Furthermore, the trigrams plus words model yields an accuracy value which is very close to the best value (84.15% against 85%).

The outcomes of the KNN classifier is illustrated in table 5. This algorithm provides the best results when it is used with the appropriate parameters (number of $k = 9$ and the measure of similarity = cosine).

The results of the table 5 indicate that utilizing KNN combined with the (trigrams plus words) model gives the highest four values of performance measurement.

From these tests we can conclude that the best representation of the tweet is the words combined with their tri-gram of characters in the three classifiers. For the algorithms used, we can say that SVM improves on the performance of NB and KNN.

Since the representation of (3-gram characters plus words) gives the best results in the these classifiers, we use this model for the study of impact of the SVM feature

Text models	Accuracy	Precision		Recall	
		Neg	Pos	Neg	Pos
words only	80.60	86.69	76.24	72.30	88.90
light stem	81	84.83	77.93	75.50	86.50
Khoja's stem	77.60	81.22	74.73	71.80	83.40
3-grams	81.70	78.82	85.22	86.70	76.70
3-grams + words	84.15	82.37	86.14	86.90	81.40
4-grams	84.20	84.41	84	83.90	84.50
4-grams + words	85	85.57	84.45	84.20	85.80

Table 4. Performance rate of Naïve Bayes on 2000 tweets dataset

Text models	Accuracy	Precision		Recall	
		Neg	Pos	Neg	Pos
words only	83.15	84.71	81.72	80.90	85.40
light stem	83.95	89.07	80.02	77.40	90.50
Khoja's stem	84.15	90.80	79.36	76	92.30
3-grams	86.20	90.04	83.03	81.40	91
3-grams + words	86.65	89.97	83.84	82.50	90.80
4-grams	86.45	90.10	83.41	81.90	91
4-grams + words	86.15	89.77	83.13	81.60	90.70

dataset ($k=9$, similarity =cosine).

Table 5. Performance rate of KNN on 2000 tweets dataset

# Features	1000	2000	2500	2600	3000	4000	5000
Precision	91.7	94.31	94.69	94.68	94.57	94.16	94.15
Recall	91.05	94.15	94.50	94.50	94.40	94	93.95
Accuracy	91.05	94.14	94.50	94.50	94.40	94	93.95

Table 6. Performance rate of SVM with (3-grams+ words) model according to the number of features

# Features	1000	2000	2500	2600	3000	4000	5000
Precision	92.14	92.98	93.34	93.14	93.03	92.82	92.20
Recall	91.80	92.80	93.20	93.05	93.00	92.80	92.20
Accuracy	91.80	92.80	93.20	93.05	93.00	92.80	92.20

Table 7. Performance rate of NB with (3-grams+ words) model according to the number of features

selection technique (SVM) on the three classification algorithms (SVM, NB and KNN). The obtained results of the effect of feature selection on the classifiers are illustrated in table 6, 7 and 8.

Table 6 gives the performance percentage of SVM according to the number of features. We can see that the best results of the three metrics are obtained with the number of features equals 2500. Compared with the results listed in table 3, the improvement of accuracy is clear (5.05%).

Table 7 illustrates the outcomes for the NB algorithm against the number of features. Similarly, NB reaches its best performance when using the first 2500 relevant features. These results demonstrate that the contribution of the SVM feature selection technique is very significant (accuracy = 93.20% against 84.15% when using all the features).

In table 8, the rates of performance measurements are given for KNN. From this table we can see that selecting only the 3000 most representative features provides the best results in two metrics (recall=accuracy= 91.50%, while the precision value 91.56% is very near to the best one 91.73%). Using the SVM feature selection technique increases the accuracy rate from 86.65% to 91.50%.

We conducted the same series of experiments on the second dataset (BBN). Table 9 shows the accuracy values of the three algorithms for the different tweet representation. As it can be seen from table 9, the baseline vector (only words) gives the worst results in the three classifiers. For this model, the best result of accuracy is provided by SVM (66.45%). We can also note that stemming methods (light and root stemmer) increased the accuracy rate for the three algorithms. Comparing the stemming methods, light stemmer gives better accuracy than root stemmer for all classifiers. The representation of (3-grams with words vector) yields the best results for KNN (68.60%). This model is competitive and gives a performance very near to the best percentage given by SVM combined with light stemmer (68.69%). Finally, 4-grams combined with words vector provided the highest accuracy rate for the NB classifier (64.13%).

To summarize the results of table 9, KNN and NB reached their best performance when using 3-grams and 4-grams plus tokens (words) vector respectively. For SVM, it provided its best accuracy with light stemmer. The trigrams

of character with words provided the second best result for SVM.

The next step of the experiments is to study the influence of the SVM feature selection technique on the sentiment classification accuracy. The optimal set of features is determined experimentally by varying the number of features and calculating the accuracy value for each set of features. The obtained results are shown in table 10.

We can note that using the first 1000 features is sufficient to achieve the best accuracy for SVM and NB. We can also say that NB is the best classifier which reaches the rate 92.92%. In the second rank comes SVM (88.72%), while KNN provides the lowest performance (73.26%).

From these experiment results conducted on two collections of tweets we can conclude that the preprocessing task is crucial for tweet sentiment classification in Arabic. Furthermore, the choice of text representation affects the performance of the classifiers.

The different n-gram models proposed in this study are better than the two stemmers (root and light based stemmer). The representation text (tri gram plus words) combined with SVM for the first data set (2000 tweets) outperforms the results of the work of [31], which used only the two famous stemming techniques (light and root stemmer). According to their study in [31], the best results are obtained when using the light stemmer and the SVM classifier (accuracy = 87.20 %).

Concerning the impact of light stemming and root stemming on classification performance, as expected, the light stemmer outperforms the root based stemmer because it tries to keep the meaning of the word by removing only some prefixes, while the root stemmer transforms the words to their root.

On the other hand, the n-gram model is robust and can tolerate the potential textual errors such as spelling errors in tweets. In fact, the tweet content tends to be noisy and do not respect grammatical rules. Moreover, the limited length of the tweet to 140 characters, make its content poor compared with long and full texts. This restriction may affect the sentiment classification performance of tweets.

Our idea behind proposing our model is to give to the tweet a rich representation which combines the words with the trigram character. In fact, it is known that more

# Features	1000	2000	2500	2600	3000	4000	5000
Precision	90.69	90.93	90.90	90.89	91.56	91.54	91.73
Recall	90.55	90.85	90.80	90.80	91.50	90.90	91.10
Accuracy	90.55	90.85	90.80	90.80	91.50	90.90	91.10

Table 8. Performance rate of KNN with (3-grams+ words) model according to the number of features

# Features	SVM	NB	KNN
Base model (words)	66.45%	57.41%	64.59%
Light stemmer	68.69%	59.55%	67.19%
Root stemmer	67.47%	57.69%	66.45%
3 gram- char	67.48%	59.19%	67.76%
3 gram- char with words model.	68.60%	62.73%	68.60%
4 gram- char	67.85%	63.75%	66.35%
4 gram+ char with words model.	67.57%	64.13%	66.17%

Similarity cosine, k = 10.

Table 9. Performance of the classifiers on BBN tweets dataset

# Features	67	1000	2000	2500	2600	3000	4000
SVM	71.03	88.72	87.79	78.70	87.23	86.77	86.12
KNN	73.26	64.22	56.30	56.57	56.11	55.55	54.80
NB	72.70	92.92	91.52	90.68	90.49	90.87	91.33

KNN, k = 10, distance = Euclidian.

Table 10. Performance rate of the classifiers with 3-Grams+words model according to the number of features

than 80% of the verbs in the Arabic language are constructed from 3-letter roots.

Furthermore, the tests demonstrate that the selection of the most important features by the SVMs technique increases significantly the performance of the three classifiers evaluated in the experiments.

6. Conclusion and Future Work

In this study, we addressed sentiment analysis for tweets in the Arabic language. We worked on two datasets which are freely available for scientific research. Opinions expressed in these collections were about general topics such as politics, arts. This work was performed to achieve three objectives. The first goal was finding the best model representation of tweets for their polarity classification

(positive, negative). The experiments performed illustrated that light stemming is preferable than the root stemming. The obtained results also indicate that the 3-grams of characters combined with tokens of the text gives the best results for opinion classification.

The second objective was to investigate the impact of reducing the size of the dataset by selecting the most relevant features on the classification efficiency and accuracy of three well used machine learning algorithms namely; Support Vector Machines (SVM), Naïve Bayes, (NB), and K-nearest neighbor (KNN). The technique of SVM feature selection used in this paper increases significantly the performance of the classifiers utilised for the sentiment classification task.

The third aim of this study was to investigate the machine

learning algorithms in Arabic opinion mining. The results showed that SVM overcomes the other classifiers (KNN and NB) in the case of using all the features. When using the feature selection technique SVM, the classifiers SVM and NB are competitive. KNN is the worst classifier in both cases; when utilizing all features and when selecting the most representative features by SVM.

As future works, we plan to extend this research by comparing the SVM feature selection method tested in this paper with other feature selection techniques such as Information Gain, PCA, and Relief. Moreover, we intend to study other problems related to the opinion mining area, such as negation, sarcasm and irony. This might help us to improve classification. Finally, the modest size of the datasets tested in this paper motivates us to compile free large collections for Arabic sentiment analysis.

References

- [1] Manning, CD., Schütze, H. (1999). Foundations of statistical natural language processing. MIT press.
- [2] Saad, M. K. (2010). The impact of text preprocessing and term weighting on Arabic text classification (Doctoral dissertation, The Islamic University-Gaza).
- [3] Wahbeh, A., Al Kabi, M., Al - Radaideh Qasem, A., Al - Shawakfa E., AlSmadi, I. (2011). The Effect of Stemming on Arabic Text Categorization: An Empirical Study, *International Journal of Information Retrieval Research (IJIRR)*, IGI Publisher, 1 (3) 54-70.
- [4] Oraby, S. M., El-Sonbaty, Y., El-Nasr, M. A. (2013). Exploring the Effects of Word Roots for Arabic Sentiment Analysis. *In International Joint Conference on Natural Language Processing*, Nagoya, Japan (471-479).
- [5] Hmeidi, I., Al-Ayyoub, M., Abdulla, N. A., Almodawar, A. A., Abooraig, R., Mahyoub, N. A. (2014). Automatic Arabic text categorization: A comprehensive comparative study. *Journal of Information Science* 41 (1) 114-124.
- [6] Duwairi R, El-Orfali M (2014). A study of the effects of preprocessing strategies on sentiment analysis for arabic text. *Journal of Information Science*, 40 (4) 501-513.
- [7] Web. *Top 30 languages of the world*, http://www.vistawide.com/languages/top_30_languages.htm (accessed 3 August 2015).
- [8] Web. *Arabic language - Wikipedia, the free encyclopedia*, http://ar.wikipedia.org/wiki/اللغة_العربية (accessed 3 August 2015).
- [9] Khoja, S. (1999). Stemming Arabic Text, Lancaster, U.K, Computing Department, Lancaster University.
- [10] Duwairi, R., Al-Refai, MN., Khasawneh, N. (2009). Feature reduction techniques for Arabic text categorization. *Journal of the American society for information science and technology*, 60 (11), 2347-52.
- [11] Duwairi, R, Al-Refai, M, Khasawneh, N. (2007). Stemming versus light stemming as feature selection techniques for Arabic text categorization. *In: Innovations in Information Technology. IIT'07. 4th International Conference on. 75 IEEE*; 2007, 446-50.
- [12] Adamson, GW., Boreham, J. (1974). The use of an association measure based on character structure to identify semantically related pairs of words and document titles. *Information Storage and Retrieval* 10 (7) 253-60.
- [13] Ahmed, F., Nürnberger, A. (2007). N-grams conflation approach for Arabic text. *In: Improving Non-English Web Searching (iNEWS07) SIGIR07 Workshop* (39-46).
- [14] Jalam R. (2003). Apprentissage automatique et catégorisation de textes multilingues. Ph.D. thesis; Université Lumière-Lyon 2.
- [15] Millar, E., Shen, D., Liu, J., Nicholas, C. (2006). Performance and scalability of a85 large-scale n-gram based information retrieval system. *Journal of Digital Information*, 1 (5).
- [16] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*; 34 (1) 1-47.
- [17] Ding, X., Liu, B., Yu, PS. (2008). A holistic lexicon-based approach to opinion mining. *In: Proceedings of the International Conference on Web Search and Data Mining. ACM*; 2008, 231-40.
- [18] Turney, P., Littman, ML. (2002). Unsupervised learning of semantic orientation from a hundred-billion-word corpus .
- [19] Pang, B., Lee, L., Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. *In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-10* (79-86). Association for Computational Linguistics.
- [20] Gonçalves, P., Araújo, M., Benevenuto, F., Cha, M. (2013). Comparing and combining sentiment analysis methods. *In: Proceedings of the first ACM conference on Online social networks. ACM*; 27-38.
- [21] El-Halees, A. (2011). Arabic opinion mining using combined classification approach.
- [22] Bermingham, A., Smeaton, AF. (2010). Classifying sentiment in microblogs: is brevity an advantage? *In: Proceedings of the 19th ACM international conference on Information and knowledge management. ACM*; 1833-6.
- [23] Tausczik, YR., Pennebaker, JW. (2010). The psychological meaning of words: Liwcand computerized text analysis methods. *Journal of language and Social Psychology*, 29 (1) 24-54.
- [24] Mayfield, J., McNamee, P., Costello, C., Piatko, CD, Banerjee, A. Jhu/apl at trec 2001: Experiments in filtering and in Arabic, video, and web retrieval. *In: TREC*.
- [25] Mustafa, SH, Al-Radaideh QA. (2004). Using n-

grams for Arabic text searching. *Journal of the American Society for Information Science and Technology*, 55 (11) 1002-7.

[26] Maamouri, M., Bies, A., Buckwalter, T., Mekki, W. (2004). The penn Arabic tree bank: Building a large-scale annotated Arabic corpus. *In: NEMLAR conference on Arabic language resources and tools*. 102-9.

[27] Rushdi-Saleh M, Martín-Valdivia MT, Ureña-López LA, Perea-Ortega JM .(2011). Oca: Opinion corpus for Arabic. *Journal of the American Society for Information Science and Technology*, 62 (10) 2045-2054.

[28] Taghva, K., Elkhoury, R., Coombs, J. (2005). Arabic stemming without a root dictionary. *In: null. IEEE*; 152-7

[29] Tashaphyne. (2010). Arabic light stemmer, 0.2. <http://tashaphyne.sourceforge.net/>

[30] Abdul-Mageed, M., Kübler, S., Diab, M. (2012). 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. Association for Computational Linguistics*, 19-28.

[31] Abdulla, N., Ahmed, N., Shehab, M., Al-Ayyoub, M. (2013). Arabic sentiment analysis: Lexicon-based and corpus-based. *In: Applied Electrical Engineering and Computing Technologies (AEECT), IEEE Jordan Conference on. IEEE*; 1-6.

[32] Shoukry, A., Rafea, A. Sentence-level Arabic sentiment analysis. (2012). *In: Collaboration Technologies and Systems (CTS), International Conference on. IEEE*; 546-50.

[33] Mountassir, A., Benbrahim, H., Berrada, I. (2013). Sentiment classification on Arabic corpora, a preliminary

cross study. *Document numérique*, 16(1) 73-96.

[34] Azmi, AM., Alzanin, SM. (2014). Aara-a system for mining the polarity of Saudi 130 public opinion through e-newspaper comments. *Journal of Information Science*; 40 (3), 398-410.

[35] Omar, N., Albared, M., Al-Moslmi, T., Al-Shabi, A. (2014). A comparative study of feature selection and machine learning algorithms for Arabic sentiment classification. *In: Information Retrieval Technology. Springer*.

[36] Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features, *European Conference on Machine Learning (ECML'98)*, Springer Berlin Heidelberg; 137-142.

[37] O'Keefe, T., Koprinska, I. (2009). Feature selection and weighting methods in sentiment analysis. *In: Proceedings of the Australasian document computing symposium*; 67-74.

[38] Salameh, M., Mohammad, S., Kiritchenko, S. Sentiment after translation: A case-study on arabic social media posts (2015). *In: NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5*, 767-77.

[39] Zbib, R., Malchiodi, E., Devlin, J., Stallard, D., Matsoukas, S., Schwartz, R., et al. 145 Machine translation of arabic dialects (2012). *In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics*; 49-59.