

Investigation of the Feature Selection Problem for Sentiment Analysis in Arabic Language

Ahmed Nasser^{1,2}, Kıvanç Dinçer^{1,*}, Hayri Sever¹

¹ Hacettepe University, Department of Computer Engineering, Çankaya Ankara, TR

² University of Technology, Control and System Engineering Department, Baghdad, IQ
{ahmed.r.nasser1984, kivanc.dincer, hayri.sever}@gmail.com

Abstract. Sentiment analysis, which is also known as opinion mining, can be defined as the process of the automatic detection of the attitude of an author towards a certain subject in textual contents. In this study we design and implement a document-level supervised sentiment analysis system for Arabic context and investigate its performance. We use three different feature extraction methods in order to generate three different datasets (unigrams, bigrams and trigrams) from the Opinion Corpus for Arabic (OCA). In order to find the optimal number of features and to obtain the best time performance in sentiment analysis, we employ two feature ranking methods (Information Gain based and Chi-Square based) and calculate the score of each feature with respect to the class labels. This feature ranking step selects only the features that are relevant to the class labels and removes the irrelevant features that cause unnecessary processing. Hence, it helps to increase the classification performance and reduce the processing time. Finally, we evaluate the performance of three standard classifiers for polarity on the previously generated unigram and bigram based data sets, namely Support Vector Machines, K-Nearest Neighbor and Decision Tree, known by their effectiveness over these types of datasets. In our study SVM classifier has showed superior classification performance compared to the other two classifiers. Our experimentation results also prove the effectiveness of the two feature selection methods we use in order to reduce the feature space of the generated datasets and provide higher classification performance.

Keywords: Arabic Sentiment Analysis, Machine Learning, Dataset Generation, Feature Selection Algorithms

1 Introduction

Sentiment analysis, which is also known as opinion mining, can be defined as the process of the automatic detection of the attitude (positive, negative or neutral) of an author towards a certain subject in textual contents by the use of natural language processing, text analysis and computational linguistics [1]. Sentiment analysis is currently considered among the most rapidly emerging research fields due to the immediate need of processing the opinionated web contents coming from social networks and web blogs.

There are many approaches in the literature that deals with the sentiment analysis of Arabic language [2]. The supervised or corpus-based approach employs different machine learning classifiers such as Support Vector Machine (SVM) [3], K-Nearest Neighbor (K-NN) [4], Decision Tree (D-Tree) [5] on some datasets to determine the sentiment orientation of the text [6]. In the alternative unsupervised approach or lexicon-based approach, special dictionaries are used to specify the polarity of a word or sentence [6]. There are also hybrid solutions combining the supervised and unsupervised approaches and they are called as weakly- or semi-supervised approaches [7].

It may be worth stating that especially the Subjectivity and Sentiment Analysis (SSA) subject has been receiving more attention among scholars [8, 9]. The SSA research includes: class prediction (i.e. subjective or objective); polarity prediction (i.e. positive, negative, or neutral) and even some level of classification (word, sentence or document level) [8].

We can summarize the goals of our study as follows: First, generating different datasets that can be used to support supervised sentiment analysis systems in Arabic context. Second, applying feature selection (reduction) techniques to reduce the feature space of the generated datasets which in turn leads the polarity classification performance to increase. Third, observing the performance of the implemented sentiment analysis system using different feature selection techniques and classifiers on the generated datasets.

We focused on implementing a supervised sentiment analysis system at document-level where a whole document should be classified as having either positive, negative or neutral polarity. We used the OCA Opinion Corpus for Arabic [1] as the base corpus for generating unigram, bigram and trigram-based data sets by applying different feature extraction techniques. We then applied different feature selection techniques [10] on those datasets to reduce the feature spaces. The features were selected using the statistical approaches such as Information Gain [11] and Chi-Square measure [11]. For the classification, we used three standard classifiers which are SVM, K-NN and D-Tree. Finally, the performance of the implemented system was experimented on the generated datasets using different testing scenarios to obtain the polarity classification performance metrics such as accuracy, precision, recall and f-measure with respect to the number of features.

In the next section, previous studies on the subject will be presented and discussed. In Section 3 we will explain the details of generating datasets as well as the method we proposed for evaluating those datasets. In Section 4 different test scenarios that are executed to evaluate the classification performance of the proposed system will be presented. In the last section we will finish the paper with the conclusion and presentation of the planned future work.

2 Related Work

Rushdi-Saleh et al. [1] proposed a document-level supervised sentiment analysis approach. They generated the Opinion Corpus for Arabic (OCA) from some movie and film reviews available on the web and used two classifiers (SVM and NB (Naïve Bayes)

[12]) to identify the polarity of those reviews. They generated three different n-gram schemes as features (unigrams, bigrams, and trigrams) and used TF-IDF (Term Frequency-Inverse Document Frequency) and TF (Term Frequency) weighting schemes in the validation process of a SVM-based sentiment classifier.

Shoukry and Rafea [13] used sentence-level supervised sentiment analysis approach for Arabic language by collecting the required data from Twitter. In the preprocessing phase, unigram and bigram based feature extraction techniques were applied and they concluded that the bigram model did not enhance the classification performance. They followed a corpus-based approach where SVM and NB with term frequency features were used for polarity classification. They extracted all the unigrams and bigrams in the corpus that exceeded a certain threshold.

Mountassir et al. [14] presented three methods to solve the imbalance issue in the dataset during SSA process: eliminate farthest, eliminate similar and eliminate by clustering. They used a document-level supervised SA approach. Two types of imbalanced corpus were generated from two different datasets: The Arabic dataset was collected from Al-Jazeera's website and the English dataset was collected from SINAI. They used a bags-of-words features representation with binary weighting. The proposed under-sampling techniques were applied on commonly used classification techniques such as NB, SVM, and K-NN and the results were comparatively analyzed.

Ahmed et al. [15] presented some of the challenges and issues that faced the SSA researchers in general and especially while working with the Arabic language, and proposed some solutions for them. They used a sentence-level supervised SA on the data collected from Twitter. They used different classifiers such as SVM, NB, Bayes Net [16] and J48 [17] to determine the polarity of Arabic tweets collected from different domains. They also tested the effects of different preprocessing techniques, feature extraction, and stemming methods on polarity classification. They used NB classifier with words N-grams frequency vector for sentiment classification.

Abdulla et al. [18] studied the two main approaches of sentiment analysis, namely sentence-level supervised and unsupervised SA for Arabic corpus collected from Twitter. They conducted different experiments using four well-known classifiers: SVM, NB, D-Tree, and K-NN. They used a sentiment lexicon to find the sentiment orientation of Arabic words and used the unigram technique for feature extraction. To determine the sentiment polarity of the entire input text, they aggregated the total sentiment scores of each individual word in that text.

Abdulla et al. [19], in another work of theirs, proposed a sentence level supervised SA with a large dataset consisting of Arabic comments. This dataset was manually collected and annotated from the "Yahoo! Maktoo" social network. They presented inclusive analysis of this dataset using NB and SVM classifiers with TF-IDF term weighting technique. Some additional information such as the number of likes or dislikes as well as the gender of the author were used to enhance the classification performance.

In [20] a sentence level supervised SA approach was presented. The dataset was collected from Arabic news websites such as Al Jazeera, BBC Arabic, Al-Youm Al-Sabe'a and Al Arabiya, Constitution Facebook Page, and People's Opinion Facebook page. They divided a "Slang Sentimental Words and Idioms Lexicon" (SSWIL) of opinion words into two classes: satisfaction and dissatisfaction classes. They proposed

a Gaussian kernel SVM classifier for Arabic slang language and used it to classify Arabic comments on Facebook.

El-Makky et al. [21] built a new Arabic lexicon by merging two Modern Standard Arabic MSA lexica, namely, MPQA [22] and ArabSenti [23] with two Egyptian Arabic lexica built from Twitter. They used both the sentence-level supervised and unsupervised SA approach. An augmented lexicon-based approach was used to define the semantic orientations (SO) of the words. The lexicon-based approach depends on the presence of opinion (or sentiment) words (looked-up from a sentiment lexicon). These words expressed positive or negative sentiments. The sentiment of the tweet that results from the modified algorithm was used as a semantic orientation score which was a component of the proposed feature vector. Subjectivity and polarity classifiers were used to classify the tweets.

3 Proposed Sentiment Analysis Approach

Our proposed Arabic supervised sentiment analysis approach consists of two parts. In the first part we generated different datasets using different feature extraction methods such as unigrams, bigrams and trigrams. These datasets were built based on the OCA Opinion Corpus for Arabic [1]. The second part of the system is the dataset evaluation and the supervised sentiment analysis system. In this part two feature selection methods were used to find the optimal number of features in each dataset in order to obtain a higher classification performance. For sentiment classification we used three different standard classifiers (SVM, K-NN and D-Tree). Each part of the proposed system is discussed in details below.

3.1 Dataset Generation

We use the OCA corpus for Arabic for generating the datasets that will be used later for building the classification model in our sentiment analysis system. The OCA corpus contains 500 text files where each file represents a movie review in Arabic language. The corpus data has been collected from 15 distinct web sites. Those 500 files are divided into two categories: positive and negative where each category consisted of 250 documents. Table 1 shows some statistics related to the OCA corpus.

Table 1. Statistics for the OCA Opinion Corpus for Arabic.

	Positive	Negative
Total word count in the corpus	130,981	104,080
Average word count for each file	524	416
Total unique word count in the corpus	79,262	66,066
Average unique word count for each file	317	265
Average sentence count for each file	12	18
Max sentence count	70	278
Max word count in a sentence	668	450
Average word count in a sentence for each file	154	75

Dataset generation process shown in Fig.1, consists of three main steps as follows:

- The first step is the preprocessing stage where each document in the corpus is first tokenized into their words and the stop words are removed. Then the root of each word is found using the Buckwalter morphological analyzer's Aramorph Arabic lemmatizer [24]. Finally, a sentiment filter is applied to these terms in order to remove the terms that are not presented in ArSenL [25]. ArSenL is a large scale standard Arabic sentiment and opinion-mining lexicon built using a combination of English SentiWordnet [26] and Arabic WordNet [27].
- The second step enumerates the terms from each document in the form of unigrams, bigrams and trigrams which represents the features that are used for generation of three separate datasets later.
- The last step does feature weighting. First, *TF-IDF* [28] values are calculated for each feature (term) in each document and used to construct the vector space model. The vector space model can be represented by an $(M \times N)$ matrix, where M is total number of documents and N is the total number of features in the corpus. This matrix contains a *TF-IDF* weight for each feature. The TF-IDF weights for each one of the previously generated unigram, bigrams and trigrams features will result in a separate data set, a total of three different datasets. Table 2 shows the total number of features for these three datasets.

Table 2. Total number of features in each dataset

Dataset	Method used	Total number of features
1	Unigram	26,270
2	Bigrams	134,871
3	Trigrams	267,451

3.2 Dataset Evaluation

The dataset evaluation process for the datasets generated in Section 3.1 is shown in Fig.2. Since the generated datasets contain large feature spaces, a feature selection method needs to be employed to reduce the number of features in each dataset. Either the Chi-square or the Information Gain (IG) method is applied to rank each feature according to its corresponding class information. Information gain is used as a metric to measure the reduction in entropy for the class category (c) prediction by knowing the presence or absence of a feature (t) in a dataset [11]:

$$IG(f, c_i) = \sum_{c \in \{c_i, \bar{c}_i\}} \sum_{t \in \{t, \bar{t}\}} P(t, c) \cdot \log \frac{P(t, c)}{P(t) \cdot P(c)} \quad (1)$$

Chi-square [11] is employed to measure the lack of independence between a feature (t) and a class category (c_i) then compared to the chi-square distribution with one degree of freedom. Chi-square is defined as follows:

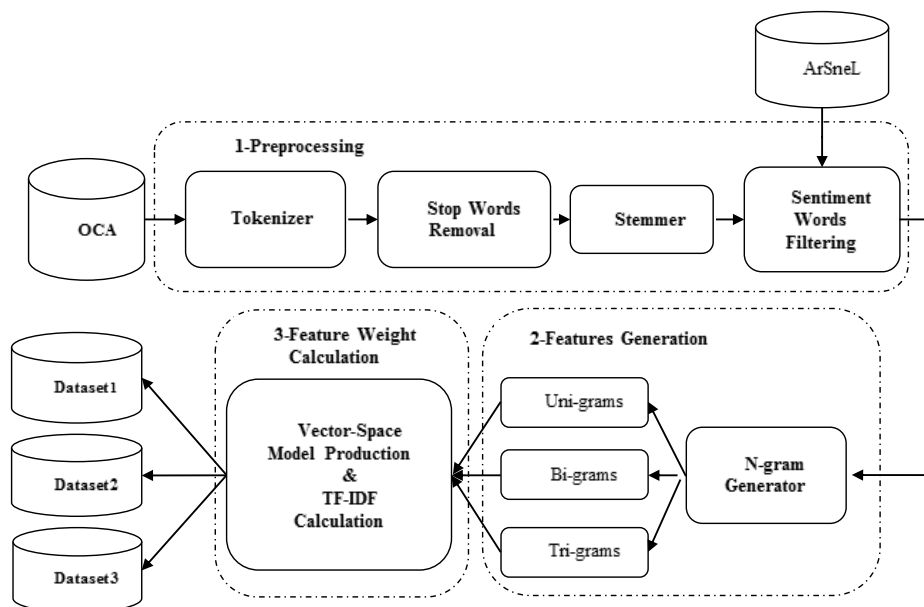


Fig. 1. Block diagram of our datasets generation method

$$X^2(t, c_i) = \frac{N[P(t, c_i)P(\hat{t}, \hat{c}_i) - P(t, \hat{c}_i)P(\hat{t}, c_i)]^2}{P(t)P(\hat{t})P(\hat{c}_i)P(c_i)} \quad (2)$$

The features are ordered separately in descending order with respect to their information gain and Chi-square values.

In the evaluation process, the dataset is divided into two sub-datasets called training and testing sets. K -fold cross-validation [29] method is followed for randomly splitting the dataset into K ($=10$ in our case) equal subsets, and each time one of the K subsets is used as the testing set and the other $K-1$ subsets are used as the training set.

The generated training set is used as input to three standard classifiers, SVM with linear kernel, K -NN with a cosine-based distance, and D-Tree. After the training process is completed, the testing dataset is applied over the trained classifiers and by comparing the results from classifiers with the class labels of the testing dataset the confusion matrix for each classifier is computed. From the confusion matrix, the classifier evaluation metrics such as accuracy, precision, recall and F-measure are calculated [29].

Before the evaluation process begins, the first 500 features (from the dataset with sorted features) are selected as input dataset to the evaluation process mentioned above and when it is done this means one iteration is complete. The number of selected features is increased by 1,000 in each iteration until the total number of features is reached.

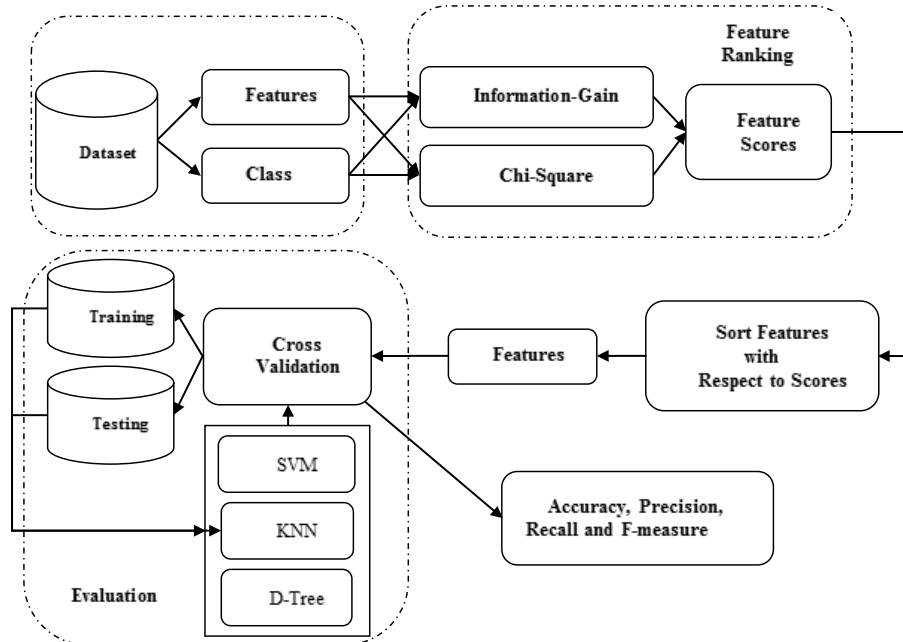


Fig. 2. The block diagram of our dataset evaluation system with the proposed feature selection approach.

4 Experimental Evaluation

At this stage we only evaluate two of the generated datasets, those that are based on unigrams and bigrams and will leave the evaluation of trigram-based dataset for the future work. The evaluation results are shown below for different scenarios:

4.1 First Testing Scenario: Testing with Unigrams

The dataset was generated from the OCA corpus using the vector space model, TF-IDF was used for term weighting, and the terms were all considered as unigrams. The total number of unigrams in the corpus are 26,270.

As stated previously, information-gain and chi-square coefficients are used to rank the features in descending order. Fig.3 shows the features sorted with respect to the information gain score.

The features are sorted in descending order with respect to their information gains and the chi-square score. The evaluation process starts with selecting the first 500 features and increasing it by 1,000 at each iteration until it reaches the total number of features. At each iteration k -fold cross validation method is used for generating the

training and testing sets, and each classifier is evaluated in 10 folds. For these 10 folds the average number of accuracy, precision, recall and F-measure metrics are calculated for each classifier. The results are shown in Fig.4 below.

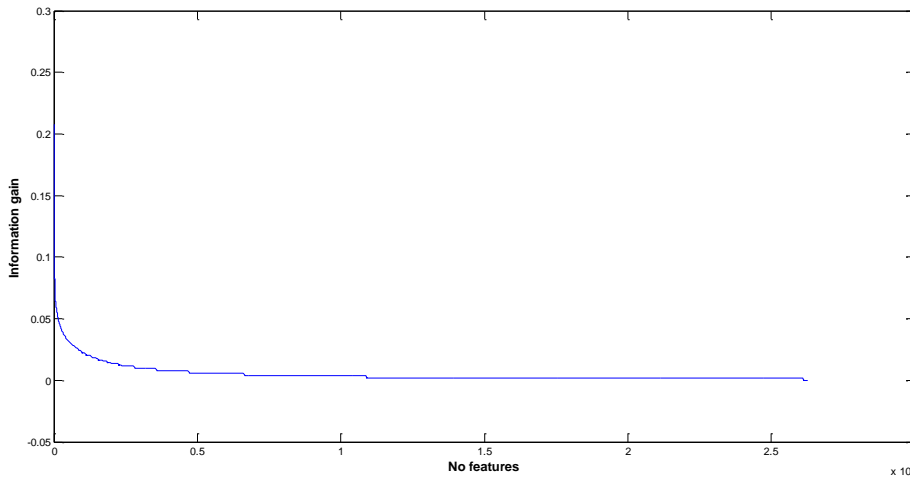


Fig. 3. Shows the features sorted with respect to its information gain score

As seen from the results presented in Fig.4, SVM classifier yields the best classification performance compared to K-NN and decision tree classifiers. D-tree classifier shows an average performance. K-NN shows the worst classification performance.

SVM classifier shows a very good performance in the region where the number of features are between 9,000 and 11,000. So the optimal number of features can be chosen as 10,000 out of the total 26,270 features and this gives 62% decrease in the feature space. Table 3 shows the performance metrics of SVM, K-NN and D-Tree classifiers with the selected 10,000 features using both information gain and chi-square score ranking.

Table 3. The performance metrics of the three classifiers using the unigram based dataset

	Information gain ranking			Chi-square score ranking		
	SVM	K-NN	D-Tree	SVM	K-NN	D-Tree
Accuracy	85	55	75	83	55	74
Precision	84	65	74	82	62	72
Recall	89	20	75	88	22	71
F-measure	88	30	75	88	34	74

4.2 Second Testing Scenario: Testing with Bigrams

The second scenario is similar to the first one with the exception that the generated dataset contains bigram based terms. The total number of bigrams for all documents in the corpus is 134,871.

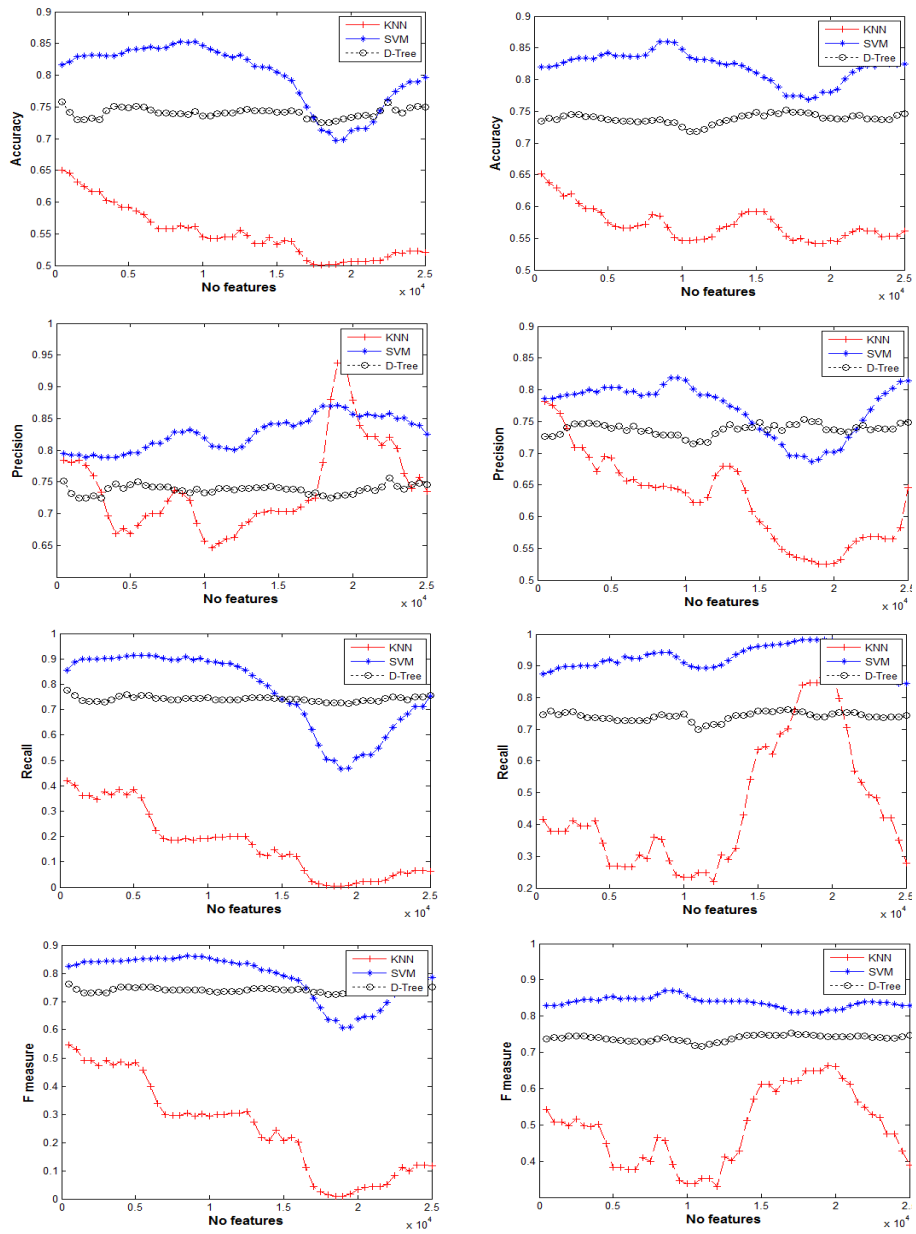


Fig. 4. (Accuracy, Precision, Recall and F-measure) versus the number of features for each classifier. The graphics on the left show the cases where the features were sorted with respect to the information gain, and the others show the features sorted with respect to the chi-square

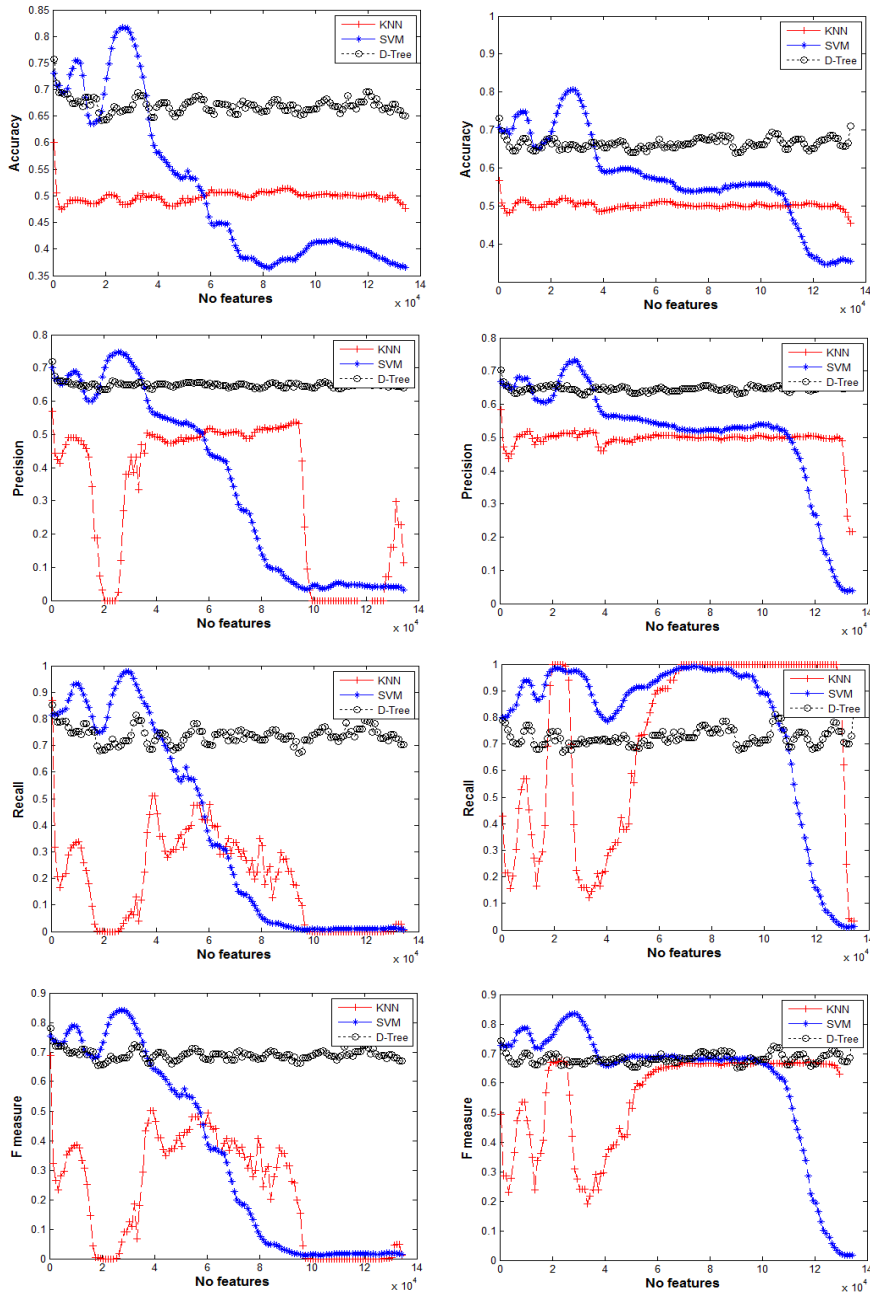


Fig. 5. (Accuracy, Precision, Recall and F-measure) versus the number of features for each classifier. The graphics on the left show the cases where the features were sorted with respect to the information gain, and the others show the features sorted with respect to the chi-square

After applying the same evaluation process described in the first scenario on the bigram based dataset, we get the classifier evaluation results shown in Fig.5.

SVM classifier shows the best classification performance among the tree, but as the number of features increases its performance drops. The D-tree classifier's performance seems to be stable with the increasing number of features. K-NN shows the worst classification performance.

SVM classifier shows the maximum performance in the region where the number of features is between 20,000 to 40,000. Therefore, the optimal number of features can be chosen as 30,000 out of the total of 134,871 features and this gives 78% decrease in the feature space. Table 4 shows the performance metrics for each classifier (SVM, K-NN and D-Tree) with the selected 30,000 features using both information gain and chi-square score ranking.

Table 4. The performance metrics of the three classifiers using the bigram based dataset

	Information gain ranking			Chi-square score ranking		
	SVM	K-NN	D-Tree	SVM	K-NN	D-Tree
Accuracy	80	49	63	75	53	64
Precision	78	50	65	70	50	61
Recall	97	15	79	96	20	65
F-measure	81	16	65	79	19	66

5 Conclusion and Future Work

In this study, we focused on implementing a document-level supervised sentiment analysis systems in Arabic context. We first generated three different datasets based on OCA Opinion Corpus for Arabic. These datasets were generated using different feature extraction methods (unigrams, bigrams and trigrams) with TF-IDF feature weighting. Then we used a supervised sentiment classification system that emphasize the Information Gain and Chi-square methods for feature selection. The value of the Information Gain and Chi-square coefficient was used for ranking of each feature in the datasets. The feature selection method aims to find the optimal number of features in each dataset that provides an optimum (higher) classification performance. Then, three of the standard classifiers (SVM, K-NN and D-Tree) were used for the sentiment classification.

We designed several testing scenarios for evaluating the performance of our proposed sentiment classification evaluation system. At each testing scenario all three classifiers and both feature selection methods were applied on each one of the previously generated datasets. We collected the (Accuracy, Precision, Recall and F-measure) classification performance metrics for each classifier using the k-fold cross validation. That is, we experimented with the generated unigram and bigram based datasets and measured the performance of each classifier (SVM, K-NN and D-Tree) using both Information Gain and Chi-square based feature selection methods.

The results show that SVM based sentiment classifier provides the best classification performance among three. The K-NN is considered as the worst classification performance, while the D-Tree classifier shows an average performance.

In our case, using the feature selection methods, the unigrams based dataset with SVM classifier shows very good performance (up to 88% in F-measure) where the number of features are approximately 10,000 out of 26,270 features. Thus this is a 62% decrease in the feature space. The bigrams based dataset shows the maximum performance (up to 81% in F-measure) using SVM classifier when the number of features is approximately 30,000 out of 134,871 total features hence, this gives 78% decrease in the feature space, providing much higher classification performance.

The advantages of SVM classifiers over the other classification methods have been reported by other researchers too. “SVM is robust in high dimensional feature spaces, works very good if any feature is relevant, data is linearly separable and most text categorization problems are considered as linearly separable” [1, 30]. It is very remarkable that SVM is superior to many other machine learning techniques [1, 13, 15, 18, 19, 20, 21].

In the future, we plan the following actions: a) Evaluating the third dataset which uses trigrams as features using the proposed evaluation system, and use Naïve-bias and neural network [31] based classifiers to evaluate the generated datasets. b) Trying other feature selection methods (such as Correlation coefficient, Odds ratio [11] and minimum-redundancy maximum-relevancy “mRMR” [32]) and showing the impact on the results of the generated datasets. c) Using hybrid features which are generated by combining more than one feature (i.e. combining TF-IDF term weighting features with term sentiment score features that can be obtained from ArSenL sentiment lexicon).

References

1. Rushdi-Saleh, M., Martín-Valdivia, M.T., Ureña-López, L.A., Perea-Ortega, J.M.: OCA: Opinion corpus for Arabic. *Journal of the American Society for Information Science and Technology*. 62, 10, 2045–2054 (2011)
2. Farra, N., Challita, E., Assi, R.A., Hajj, H.: Sentence-Level and Document-Level Sentiment Mining for Arabic Texts. In: 2010 IEEE International Conference on Data Mining Workshops. pp. 1114–1119 IEEE (2010)
3. Fung, G.M., Mangasarian, O.L.: Multicategory Proximal Support Vector Machine Classifiers. *Machine Learning*. 59, 1–2, 77–97 (2005)
4. Dasarathy, B.: Nearest Neighbor (NN) Norms : NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos (Cal.) [etc.] (1991).
5. White, A.P., Liu, W.Z.: Technical Note: Bias in Information-Based Measures in Decision Tree Induction. *Machine Learning*. 15, 3, 321–329 (1994)
6. Read, J., Carroll, J.: Weakly supervised techniques for domain-independent sentiment classification. In: Proceeding of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion - TSA '09. p. 45 ACM Press, New York, New York, USA (2009)
7. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-Based Methods for Sentiment Analysis. *Computational Linguistics*. 37, 2, 267–307 (2011)
8. Montoyo, A., Martínez-Barco, P., Balahur, A.: Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments. *Decision Support Systems*. 53, 675–679 (2012)
9. Abdul-Mageed, M., Diab, M., Kübler, S.: SAMAR: Subjectivity and sentiment analysis for Arabic social media. *Computer Speech & Language*. 28, 1, 20–37 (2014)

10. Sarakar, S., Goswami, S.: Empirical Study on Filter based Feature Selection Methods for Text Classification. *International Journal of Computer Applications*. 81, 6, 38–43 (2013)
11. Zheng, Z., Wu, X., Srihari, R.: Feature selection for text categorization on imbalanced data. *ACM SIGKDD Explorations Newsletter*. 6, 1, 80 (2004)
12. Pazzani, M., Domingos, P.: On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*. 29 (2–3). p.p 103–130 (1997)
13. Shoukry, A., Rafea, A.: Sentence-level Arabic sentiment analysis. In: 2012 International Conference on Collaboration Technologies and Systems (CTS). pp. 546–550 IEEE (2012)
14. Mountassir, A., Benbrahim, H., Berrada, I.: An empirical study to address the problem of Unbalanced Data Sets in sentiment classification. In: 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC). pp. 3298–3303 IEEE (2012)
15. Ahmed, S., Pasquier, M., Qadah, G.: Key issues in conducting sentiment analysis on Arabic social media text. In: 2013 9th International Conference on Innovations in Information Technology (IIT). pp. 72–77 IEEE (2013)
16. Friedman, N., Geiger, D., & Goldszmidt, M.: Bayesian network classifiers. *Machine learning*, 29(2-3), p.p 131-163 (1997)
17. Clark, P., Boswell, R.: *Practical Machine Learning Tools and Techniques with Java Implementation*. Morgan Kaufmann, San Francisco (2000)
18. Abdulla, N.A., Ahmed, N.A., Shehab, M.A., Al-Ayyoub, M.: Arabic sentiment analysis: Lexicon-based and corpus-based. In: 2013 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT). pp. 1–6 IEEE (2013)
19. Abdulla, N.A., Ayyoub, M. Al, Kabi, M.N. Al: An extended analytical study of Arabic sentiments. *International Journal of Big Data Intelligence*. 1, 1/2, 103 (2014)
20. Soliman, T., Soliman, T.H., Elmasry, M.A., Hedar, A., Doss, M.M.: Sentiment Analysis of Arabic Slang Comments on Facebook. *International Journal Of Computers & Technology*. 12, 5, 3470–3478 (2013)
21. El-Makky, N., Nagi, K., El-Ebshihy, A., Apady, E., Hafez, O., Mostafa, S., Ibrahim, S.: Sentiment Analysis of Colloquial Arabic Tweets. In: ASE BigData/SocialInformatics/PASSAT/BioMedCom 2014 Conference, Harvard University. pp.1–9 (2014)
22. Mihalcea, R., Banea, C., Wiebe, J.: Learning multilingual subjective language via cross-lingual projections. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. 976–983 (2007)
23. Abdul-Mageed, M., Diab, M.: SANA: A Large Scale Multi-Genre, Multi-Dialect Lexicon for Arabic Subjectivity and Sentiment Analysis. *Proceedings of the Language Resources and Evaluation Conference*. 1162–1169 (2014)
24. Buckwalter T. Buckwalter Arabic Morphological Analyzer version 1.0, Philadelphia: Linguistic Data Consortium, Catalog No.LDC2002L49, ISBN 1-58563625760, 2002.
25. Badaro, G., Baly, R., Hajj, H., Habash, N., El-Hajj, W.: A large scale Arabic sentiment lexicon for Arabic opinion mining. In: ANLP 2014, p. 165 (2014)
26. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining SentiWordNet. In: *Proceedings of the Seventh Conference on International Language Resources and Evaluation*. pp. 1–12 (2010).
27. Black, W., Elkateb, S., Pease, A., Rodriguez, H., Alkhalifa, M.: Introducing the Arabic WordNet Project. *Word Journal Of The International Linguistic Association*. 22, 295–299 (2006)
28. Sebastiani, F., Fabrizio: Machine learning in automated text categorization. *ACM Computing Surveys*. 34, 1, 1–47 (2002)
29. Mitchell, T: *Machine learning*. MIT Press. (1997)

30. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. pp. 137–142 Springer Berlin Heidelberg (1998)
31. Dhande, L. L., Patnaik, P. G. K.: Analyzing Sentiment of Movie Review Data using Naive Bayes Neural Classifier. 3(4), 313–320 (2014)
32. Peng, H. C., Long, F. H., Ding, C.: Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 27, 8, 1226–1238 (2005)