



SENTIMENT ANALYSIS FOR ARABIC AND ENGLISH DATASETS

R. M. Elawady

Communication and Computer systems Department,
Faculty of Engineering, Mansoura University - Egypt

S. Barakat

Information Systems Department, Faculty of Computers
and Information, Mansoura University - Egypt

N. M. Elrashidy

Abstract: *Sentiment analysis is an important topic that has tracked attention since 2001. It basically is text classification based on analyzing opinions that expressed by writing (e.g., social media, blogs, discussion groups, etc). The widespread use of social networks has, also, led to a widespread availability of opinionated posts, making research in the area more viable and important. We need to make sentiment analysis to calculate the percentage of user acceptance or rejection according to their comments. Although Arabic is the native language of hundreds of millions of people in twenty countries across the Middle East and North Africa, the research in the area of Arabic sentiment analysis is progressing at a very slow pace compared to that being carried out in English[2]. In this paper, we present our work in which we start by testing on English texts that were collected from Amazon (book, DVD, and electronics). Then, we applied the same process on Arabic dataset that we collect from YouTube Arabic pages. We applied more than one machine learning on algorithms both (Arabic. English) (Decision trees, Navie Bayes, functions, and support vector machines. We also created a Sentiword Lexicon based on the Corpus that we gathered. Then we evaluated each method and compared their accuracies.*

Keywords: Analysis for Arabic comments; machine learning algorithms; sentiment analysis; opinion mining

1. Introduction

Opinion mining (OM) one of the fields of data mining that based on opinions that extracted from text files, three main sections extracted from OM, sentiment classification, development of linguistic resources and opinions summarization, as shown in figure 1 [1]. Sentiment analysis (SA) is the process of classification opinions (e.g. positive or negative or neutral) from the documents like product reviews/movie reviews. internet now have huge number of users which they use internet to express their opinions about any products, movies, books, restaurants Political personalities, people use blogs, social websites, discussion groups to express their opinions, there for the demand for sentiment analysis increased. Mining of those unstructured text has become one of the most active researches with many applications. The idea of sentiment analysis to classify the opinion into positive and negative sentiment. Another idea is to give the document rate from 1 to 5. Most sentiment analysis applications

are focus on English data [5] [6] and few dataset in Arabic [4] [5] [7], we start with English data set, and due to the lack of Arabic dataset .we think about make an Arabic data set but a lot of complications make identify Arabic opinions more complex compared to huge number of available English resources and lexicons such as lack of resources and high ineffectual nature of Arabic language [7], the variant sources of ambiguity [3] [8], and rich metaphoric script usage remain the most challenging problems

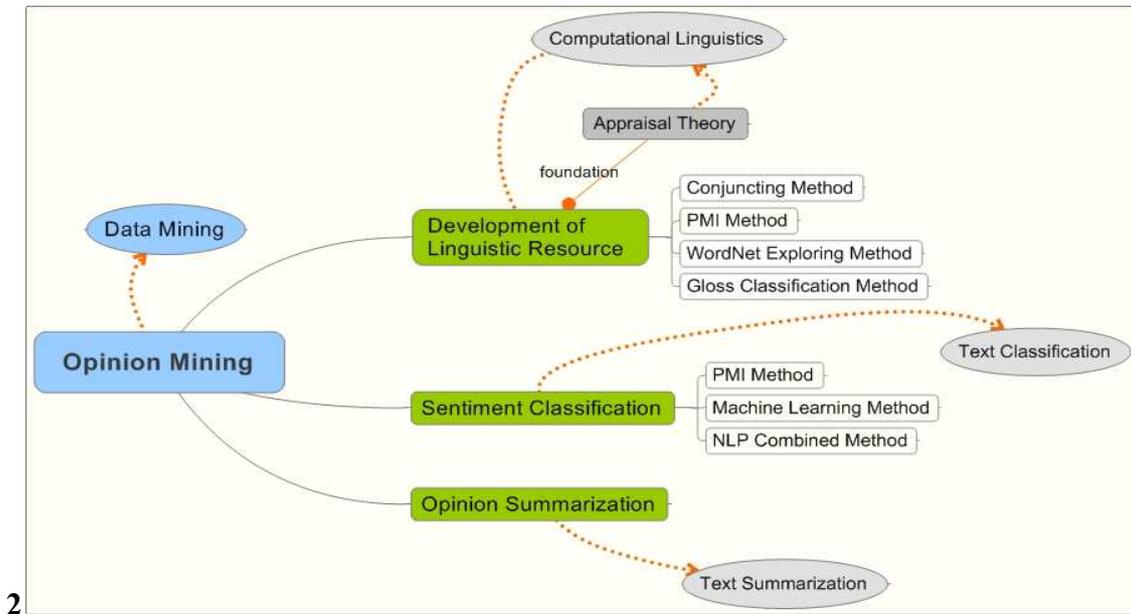


Figure. 1.1 Relationship between opinion mining and its tasks

In this work we try to summarize all the experiments that we previous do both in English and Arabic data which address lack of Arabic sentiment analysis datasets in this field, in the hope of sparking more interest in research in Arabic sentiment analysis and related tasks. Towards this end, we make the following contributions:

- Compare the performance of different machine language (ML) based on English data set
- Create new Arabic dataset based on the opinions on the movies from YouTube.
- Show that that information gain (IG) is an effective feature selection method, not only in text categorization, but also in sentiment classification.
- Show that NB give better results in classification for movie reviews
- Create SentiWord lexicon based on the Arabic dataset
- Compare the performance ML and SentiWord Lexicon that we developed and show what is better in text classification.

The paper is organized as follows: Summarization of related work given in Section 2. Feature selection methods for choosing most important features discussed in Section 3. Section 4 explain machine learning algorithms. Section 5 show testing algorithms in English data set. Section6 introduces challenges with Arabic data .Corpus that we were creating discussed in section 8 SentiWord lexicon described in section 9, Experimental setup and results are discussed in Section 10. Finally, Section 11 describes conclusions.

2. Related Work

Sentiment classification aims to distinguish whether people like/dislike a product from their reviews. It has emerged as a proper research area. AhmedAbbasi et al., [4] developed sentiment analysis technique to classify opinions that write in text documents. The developed system for sentiment analysis utilized the function of stylistic and syntactic features to estimate sentiment in both Arabic and English. Elhawary and Elfeky in [9] proposed a method to extract reviews that were written in Arabic language, after that reviews are analyzed and determine its sentiment, they use Arabic Similarity Graph which is lexicon based method. No evaluation has been made by this work. Almas and Ahmad in [12] develop method for extracting high quality terms called as local grammar automatically for Arabic and English language they used data from financial news. However, they have reached an acceptable precision of (85.1 %) the performance of the method is very low especially for the recall which is about (17.2%). The Entropy weighted Genetic Algorithm is incorporated to enhance the performance of the classifier and achieve the true assessment of the key features. Experiments were conducted using movie review data set and the results demonstrated that the proposed techniques are efficient. However, there domain is very specific since hate and extremist vocabulary is limited and it is not hard to distinguish positive and negative words. Also, they did not use any preprocessing stage which is crucial for Arabic language.

3. Techniques Used in This Study

3.1 Problem Definition

Sentiment analysis aims to determine the opinion in such a product and if its satisfy customer prediction or not for future improvements we try to build a model that can classify a comment to positive or negative .We try to make a solution that give us high accuracy by minimum number of features of the original and machine learning method are used to make calculate classification accuracy.

3.2 Information Gain (IG)

Information gain (IG) is one of the main feature selection methods that used in sentiment classification .IG select the top ranked features according to class. Feature selection use to reduce the features size. So when the feature is pre identified the important features choices, and it gives better accuracy in classification.

$$IG(t) = - \sum_{j=1}^K P(C_j) \log(P(C_j)) +$$

$$P(w) \sum_{j=1}^K P(C_j | w) \log(P(C_j | w)) +$$

$$P(\bar{w}) \sum_{j=1}^K P(C_j | \bar{w}) \log(P(C_j | \bar{w}))$$

Where , $P(C_j)$ the total number of text files that classified to class C_j out of total text files, and $P(w)$ total number of text files in which w term appears. $P(C_j|w)$ the total number of text files from class C_j that have term w .

4. Machine Learning Algorithms

4.1 Multinomial Naive Bayes

Naive Bayes [15] is one of the machine learning algorithms used especially for text classification, its one the efficient algorithms due to its efficiency and usability, it works on conditional independent. [15]. A Multinomial Naive Bayes classifier [13] with term frequency. Build based using word count or term existence in the text file, In Boolean Multinomial Naive Bayes (BMNB) [13], word counted 1 if it exist and other it counted zero

4.2 Support Vector Machine (SVM)

Support vector machine one of the most efficient supervised learning methods, this method .SVM| try to find the maximum hyper plane that can separate the training document in way that be highly separable. SVM has used mainly for classification text files [15] and for sentiment analysis [10], [11]. Much like the human brain, SVMs learn by example. Each example consists of an m number of data points (x_1, x_2, \dots, x_m) followed by a label (or target), which in the two class classification we will consider later, will be +1 or -1. -1 representing one state and 1 representing another.

4.3 Decision Trees

The tree method used to calculate decision tree. It is simple to test and in editing, and this tree can be used to extract set of decision rules. this approach are valuable in case we have different number of attribute in a huge database. Decision Tree classification are simple especially in identification of significant variable because the result create as a binary tree like a structure

5. Sentiment classification with English Data set

5.1 Dataset

For testing our method we use four dataset collected from Amazon.com (book, DVD, and electronics) each contains 2 reviews 1000 positive and 1000 negative.

5.2 Evaluation Metrics

The performance of most classifiers is measured using the popular metrics precision, recall, f-measure and accuracy. First precision express the correctness of the used classifier it means that when the precision increase the number of false positive decrease. (False positive means the documents that is negative or neutral and classified as positive incorrectly). On the contrary, lower precision means the number of documents classified as false positive increase. Recall considers the completeness of the precision, as precision increase recall decrease. Recall expresses the sensitivity of the specified classifier.

The accuracy expresses the correctness of each classifier in classification documents. The f-measure expresses the average of the both, f-measure gives high result when the two values are near.

$$F\text{-Measure} = 2 * (\text{Precision} * \text{recall}) / (\text{Precision} + \text{recall})$$

5.3 Different feature Selection methods For Sentiment analysis

We use more than method for feature selection but every one of them have advantages and disadvantages. First IG can use to specify the percentage of importance of every term. This method also used to determine redundancy in data and also give high number of attribute if the data is huge. mRMR use to ignore feature that considered redundancy. mRMR has a merit that it take in its account the dependency between features and also it save time so when we combine RS and IG it give better. These approach first calculate information gain for every attribute and choose all have IG more zero, so if this step we eliminate all redundant and noisy this step take effort and assumptions. Then send this features to RS to get the optimal feature. This method show that combine these method is perform better and save effort and time.

6. Results and Discussions

Firstly, we extract the unigram features and choose all features and ignore any feature selection algorithms, then we apply feature selection techniques to choose the best features that give the best result. We find that IG and mRMR consider the best in feature selection method in this case. Product reviews are used that contain three data set (DVD, Books, Electronics, are used after some preprocessing. Table 6.1 shows the result of every feature selection method we used it first show number of unigram features and then rough set features and features using mRMR. And finally number of features when we used Rough set and information gain which give better performance.

Table 1: Feature Length for Various Features Used with Four Dataset

	Movie	Book	DVD	Electronics
Unigram F	9045	5391	5955	4270
Rough F	263	310	350	371
IG-F	263	310	350	371
mRMR F	280	360	320	380
Hybrid F	330	410	403	405

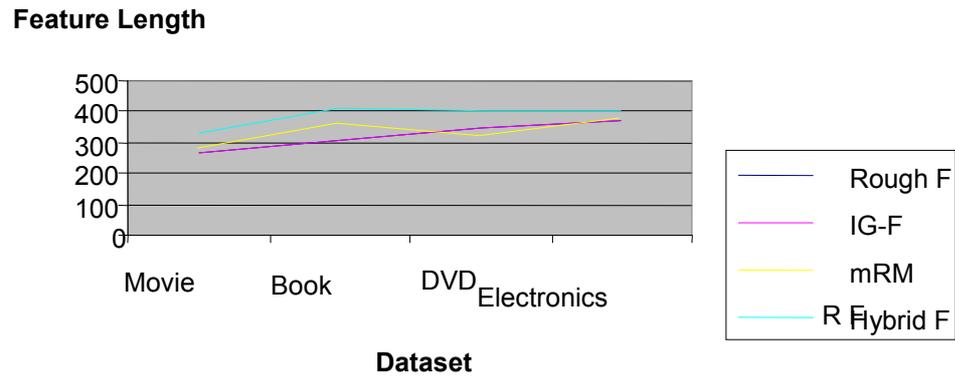


Figure.2. Number of features and algorithm

Table2 F-measure(in%)for various features with four dataset

	Movie		Book		DVD		Electronics	
	SVM	NB	SVM	NB	SVM	NB	SVM	NB
Unigram-F	84.2	77.1	76.2	74.4	77.3	74.2	76.5	74.5
Rough -F	85.6	78.1	78.0	74.9	80.4	76.5	82.9	75.5
IG-F	85.9	78.6	77.0	76.3	79.1	75.1	81.1	75.2
mRMR	86.0	79.4	79.3	78.3	79.2	77.2	82.7	79.0
Hybrid- F	87.7	80.9	80.2	79.1	83.2	78.1	83.5	78.1

Experimental shows that mRMR and IG selects about 60-75% of common features, mRMR able to remove noisy features that unwanted. It can select only optimal features and retains only features with minimum correction among in all features. When we use hybrid (rough set+IG) method make the accuracy increase about 1.5% and when use (Rs+ IG+mRMR) the accuracy increase about 4.2% that's because RS use to remove redundant data then IG choose the top features then send to mRMR to eliminate unused features.

7. Challenges of Arabic Sentiment

The main difficulty of performing sentiment analysis in Arabic social media lies in the fact that communication within the social media context is carried out using “spoken” or colloquial Arabic rather than the more formal Modern Standard Arabic (MSA). Not only is the vocabulary of colloquial Arabic different than that of MSA, the structure of the sentences is much more random which is why parsing this text poses a major challenge [2].

Lexicon: The first step in the data that used in sentiment analysis and classification, is to determine the polarity of the given text whether it positive, negative or natural by using sentiment lexicon. Sentiment lexicon is typically consisting of a list of words and their trends as either negative, positive terms, but this lexicon is rare. The first lexicon developed on Arabic mining is that presented in [3] in which the

goal was to mine Arabic business reviews. In this work, the authors used a seed set of 1600 words (600 positive, 900 negative, and 100 neutral). The seed words were used within an Arabic similarity graph built using a large web corpus to determine the polarity. Abdul-Mageed et al [16] presented a system for carrying out sentence level subjectivity and sentiment analysis for modern standard Arabic (MSA) text. Abdul-Mageed and Diab also presented a system for learning an Arabic lexicon [4], but again, the focus was on MSA terms

8. Arabic Dataset

In this section we start by describing the needed pre-processing phase to the input Arabic text. The details of these approaches that we used in build classification model are given in sections 8 and 9 and the results analysis is given in section 10.

8.1 Corpus Collection and Preparation

Arabic NLP resources are rare, therefore we had to build the needed resources, and perform the required pre-processing on it. The pre-processing tasks include sentence segmentation, morphological analysis, part of speech tagging (POST), and semantic analysis.

We downloaded over 28300 reviews from YouTube social network www.youtube.com during the month of November 2013. These reviews were from the first 214 Movie in the list of Arabic Movies. After harvesting the reviews, we found out that over 25% of them were not in Arabic and 3% written in Franco Arabic. After filtering out the non-Arabic reviews, and performing several pre-processing steps to clean up HTML tags and other unwanted content, we ended up with 200000 Arabic reviews.

8.2 Preprocessing

After we collected from YouTube, we striped out the HTML tags and non-textual contents. After that we divide these comments into sequential text files arranged in sequential order. Then begin in the first step in preprocessing which is to normalize the data (normalize all letters that have more than one writhing method) then make stop words removing and stemming.

8.2.1 Collecting Data and Dividing

Data in this context is data that contains negative and/or positive. We decided to collect negative and positive Comments at the same time instead of collecting them in different days. Collecting them at the same time was considered good because it can capture some subtle differences between negative and positive posts in the same time. we followed Some preprocessing steps in order to reach to total initial data .First, After data collected, A total of 30,545 Comment were filtered and divided into two classes (positive and negative) according to lexicon. We used enhanced version of Samhaa lexicon [5] it contains 4000 term but we increase it with some vernacular terms that we will used in our dataset Each of the terms in the lexicon was tagged as being either a verb, adjective, noun, adverb, or idiom/compound

8.2.2 Non-Arabic removal

Both positive and negative data were separated into Arabic and non-Arabic data. This was possible by using regular expression in C# and build our own code to loop to all files and ignore files that contain letters {A-Z, a-z}.

8.2.3 Stop-words removal

Second, stop words are removed from the data we collected Arabic stop words we create lexicon for Arabic stop words collected from multiple domains and create our own code to remove this words from comments. The list of words that we collected is called a stop word list. Stop words are not useful in Search. Stop words remove to save time and space, we build our own code that take the list of words and remove it.

8.2.4 Stemming

After removing stop-words, the sentence is tokenized, and stemming is applied using ArabicStemmerKhoja. For example, when the word “الاقتصادي” is stemmed to its root the returned token is “قصد”. In this example, the root is very different from the original and can be the root of many other words which can have very different meanings such as: مقصود، قاصد. Stemming have two main disadvantages, first, more than one words may have the same root. Second, poor stemmers not have all the words.

9. SentiWord-lexicon Approach

In this approach, we proposed a machine learning based senti-word lexicon based on YouTube comments that we gathered before. Building manual lists may take much time and effort. To overcome these issues, we suggest another approach that uses a senti-word Lexicon where each term is associated with its sentiment along with a weight. These weights are extracted using SVM based on the comments of YouTube. We divide this process into three phases: data preparation, lexicon development, and the classification step that shows the results of using this lexicon.

9.1 Data Preparation Phase

After data collected and divided in to files and make the preprocessing steps we previously made (classifying, stemming, stop words removing, normalization). We use this data in building lexicon.

9.2. Lexicon Development phase

First, we built our own program to extract the unigram features and bigram features. Mixing of unigram features and bigram features are using with some weight, we consider the bigram features has a high weight in our lexicon we give high priority for the bigram over the unigram

After we extract the words and their frequency we start in calculating weight for every term frequency and idf (inverse document frequency). And the result of multiply then out the weight for every term. Here we have term frequency which express number of appearance each term in a document and to calculate the importance of every word in a document we use this equation

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

where $n_{i,j}$ is the number of appearance of each word (t_i) in document d_j , and the output express number of appearance of all words in all documents. Then inverse document frequency express the degree of importance of every word by this equation

$$idf_i = \log \frac{|D|}{|\{d_j : t_i \in d_j\}|}$$

Where D the summation of all documents in the dataset and $[d_j : t_i \in d_j]$ all document where word appears

$$Tf = tf * idf$$

The output of this stage is a CSV file which contain the unigram and bigram Features and its frequency, then we send this file to LibSVM to calculate the weight for every term for this (we use Octave free software) after changing the CSV file to LibSVM file using Weka software. After this step we reach to CSV Lexicon File contain the terms with its weight that is used for classification further

Classification phase

We used weights calculation methods to apply senti-word lexicon (SWL) for reviews classification. The classification process done by take the summation of weights either positive or negative words, the signal of this summation specify the polarity of the comment

Evaluation Measure

We calculate accuracy by dividing number of correctly classified comments by the total number of comments, as in the following equation:

$$accuracy = \frac{correct}{processed - unclassified} \times 100$$

10. Experimental setup and result

10.1 Results with Machine Learning

Experiments with IG feature selection method using the SVM and Naive Bayes classifier has shown that, in general, according to the f-measure, IG performs better than the other feature selection methods in most text categorization tasks [25]. Here our objective is to experiment using IG and for sentiment classification on SVM Decision Trees and Naïve Bays classifier to ascertain if this conclusion will still hold. The results of our experiments are presented in the following sections

10.1.1 Initial classification without feature selection

Table 3 shows the summary of average 10-fold cross validation accuracies in percent for the base classifier. Naïve Bays performed better than SVM accuracies for both classifiers were reduced. SVM had an accuracy of 83.96% and Naïve Bays had an accuracy of 89.34%.

Table 3 Average 10-fold cross validation accuracies for base classifier in percent

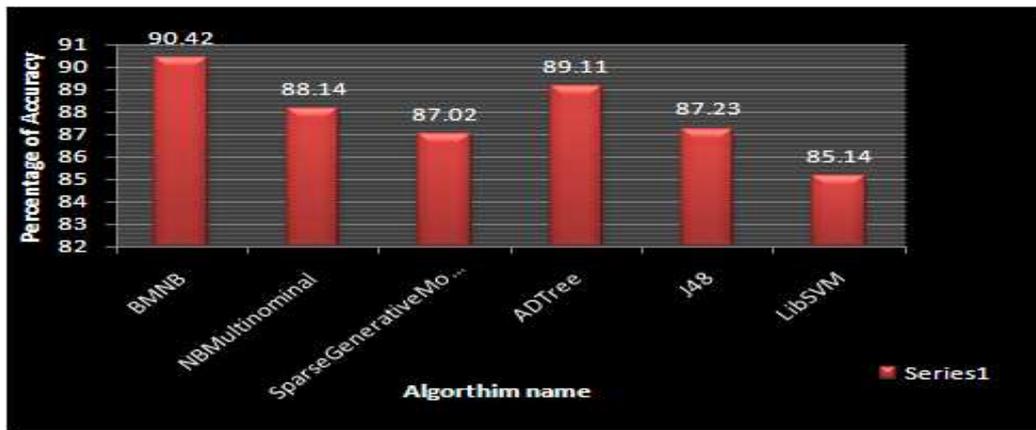
Dataset	Number Of Features	SVM	NB
YouTube	14,657	83.96%	89.34%

10.2 Results with feature selection

Table 4 :results of machine learning with feature selection

Classification Technique	Correctly Classified Instances	Incorrectly Classified Instances	TP Rate (AVG)	FP Rate (AVG)	Precision (AVG)	Recall	F-Measure	Classifier Name
Naïve Bayes	90.4286%	8.5714 %	0.914	0.038	0.915	0.914	.924	BMNB
	88.1429%	11.8571 %	0.885	0.025	0.872	0.871	.842	NBMultinomial
	90.1429%	9.8571 %	0.901	0.029	0.922	0.901	.901	SparseGenerativeModel
Decision Trees	89.1429%	10.8571 %	0.821	0.112	0.821	0.821	.892	J48
	87.1429%	12.8571 %	0.810	0.106	0.810	0.811	.863	ADTree
Functions	85.1429%	14.9286 %	0.851	0.113	0.842	0.831	.821	LibSVM

Experimental shows that IG performs considerably better with optimal features for classifying instances compare to results reported in previous literature.



Tables 4 show the summary results for the YouTube dataset generated when feature selection is performed during the classification stage using SVM Decision Trees and Naïve Bayes, respectively.

The first and the second column represent number of reviews that correctly classified and number of reviews that in correctly classified the 2 columns after this represent the rate of True positive (TP rate) and the rate of False positive (Fp Rate) then the precision and the recall and F-measure that were discussed in section 8.1

Table 3 represents the best results generated without feature selection method, Table 4, we notice that IG performs better and the Accuracy of classification increase about 3-5% in each model. Its best accuracy results of 91% were obtained with only 42% of the features.

10.3 Results with senti word-lexicon approach

We make 16 experiments which is combination of testing (Normalization, stemming, removing stop words, weights) and show what will result high performance and accuracy

- We conducted experiments with four varying factors: stop words, stemming, normalization, and weights.
- For experiments which didn't make use of the weights factor (1, 3, 5, 7, 9, 11, 13, and 15), we found that using normalization and stop word removal achieved best accuracy (93.822%). The accuracy declined slightly to 93.112% when only normalization was used.
- Using the stemming factor has affected that accuracy of the results negatively by 4-5% depending on other factors used (as in experiments 5, 7, 13, and 15). We attribute this decline in accuracy to the fact that the stemmer used was for Modern Standard Arabic, not Colloquial Arabic which the data mostly consisted of. In addition, the stemmer can produce one stem for multiple tokens that vary in their negativity.
- Removing stop words affected the accuracy positively by .9% with an increase in program speed. As in experiments 3, 7, 11, and 15.

Table 5: Results of Senti-word lexicon

Experiment No.	Normalized	Stemmed	Removing SW	weights	Accuracy
1	NO	NO	NO	NO	93.112%
2	NO	NO	NO	Yes	94.574%
3	NO	NO	Yes	NO	93.728%
4	NO	NO	Yes	Yes	94.513%
5	NO	Yes	NO	NO	85.544%
6	NO	Yes	NO	Yes	87.987%
7	NO	Yes	Yes	NO	86.756%
8	NO	Yes	Yes	Yes	87.454%
9	Yes	NO	NO	NO	93.822%
10	Yes	NO	NO	Yes	95.731%
11	Yes	NO	Yes	NO	93.912%
12	Yes	NO	Yes	Yes	95.936%
13	Yes	Yes	NO	NO	85.643%
14	Yes	Yes	NO	Yes	85.934%
15	Yes	Yes	Yes	NO	86.432%
16	Yes	Yes	Yes	Yes	87.934%

- In cases where weights are not used, classification is done as follows: the sentence (or the comment) is analyzed by calculating the number of negative and positive words, according to the prebuilt lexicon. The larger of the two determines the sentiment of the sentence.
- The rest of experiments, where weights are factored in, the classification process is performed by calculating the weight of each word (as explained before in section 7.3). Then, the weights of all words are summed and the sign of the calculated value determines the sentiment of the sentence.
- The weights factor by itself increased the accuracy of classification by 1%, as it was the case with experiment 2.
- The best results were observed in experiment 10 when using both weights and normalization.
- Combining stop word removal and weights increased the accuracy slightly as explained before, as in experiments 4, 8, 12, and 16.
- The highest accuracy (95.936%) obtained when used (normalization & stop words removing & weights) in experiment 12 and the lowest (85.544%) when using stemming only in experiment 5

10.4-Comparison with other works

From Literature review can notice that the accuracy of most methods in most domains is low . That is mainly because of the complexity of the Arabic language. In order to understand the advantages, disadvantages, and limitations of the various sentiment analysis methods, we present comparison results among them. We begin by comparing the coverage of all methods of sentiment analysis using machine learning and then using senti-wordnet

Table 6:Comparison other work

Paper	Dataset size	Dataset type	Method	Preprocessin g	Accuracy
Samahaa [5]	(Twitter dataset) 500 tweets	Arabic	Weighted Lexicon	No	82.45%
A..Hamoud [17]	Facebook comments	Arabic	ML (NB, SVM)	Yes	80.12%
A. Mourad [6]	Microblogs (such as Twitter)	Arabic	SSA Lexicon	Yes	83.54%
A. El Halees [40]	Multiple domain	Arabic	Maximum Entropy K-nearest	No	84.3%
A.Aly [41]	Book reader	Arabic	BNB, MNB ,NB	No	90.1%
Our work	YouTube Comments	Arabic	NB,SVM,DT,SWL	Yes	94.5%

This is support our suggestion that it is better to use combined methods to classify Arabic documents. Table 5 gives the accuracy of applying combined methods.

Conclusion

In this research we used the corpus that collected from YouTube for sentiment analysis purpose. YouTube are used to collect both the training and the testing data that is collected in the same time ,different feature selection methods are used and chose the one that give the best accuracy, more than one machine learning technique are applied, we find that NB five the high accuracy which reached to 91.4% , .we develop sentiword lexicon based on the corpus we collected from YouTube and used this lexicon for classification data which reach accuracy about94.5 %

References

1. William BLACK, Sabri ELKATEB Piek VOSSSEN, "Introducing the Arabic WordNet Project "Sci&Eng."
2. Samhaa R. El-Beltagy, Ahmed Ali, "Open Issues in the Sentiment Analysis of Arabic Social Media: A Case Study",
3. A.A. Hamoud, F.A. El. Taher, "Sentiment Analyzer for Arabic Comments System", (*IJACSA International Journal of Advanced Computer Science and Applications, Vol. 4, No.3, 2013*)
4. Ahmed Abbasi, Hsinchun Chen, and Arab Salem. 2008. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Transactions on Information Systems (TOIS)*.
5. Muhammad Abdul-Mageed and Mona Diab. 2012a. Awatif: A multi-genre corpus for modern standard Arabic subjectivity and sentiment analysis. In *Proceedings of the Eight International Conference on Language Resources and Evaluation*.
6. Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2:1–135
7. Mohammed Korayem, David Crandall, and Muhammad Abdul-Mageed. 2012. Subjectivity and sentiment analysis of Arabic: A survey. In *Advanced Machine Learning Technologies and Applications*.
8. M. Rushdi-Saleh, M. Martín-Valdivia, L. Ureña-López, and J. Perea-Ortega. 2011a. Bilingual experiments with an Arabic-English corpus for opinion mining. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*.
9. M. Elhawary, and M. Elfeky. (2010). "Mining Arabic Business Reviews." *IEEE International Conference on Data Mining Workshops*.
10. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 79–86 (2002)
11. Yousif Almas and Khurshid Ahmad. A Note on Extracting 'Sentiments' in Financial News in English, Arabic & Urdu. In *Proceedings of the 2nd Workshop on Computational Approaches to Arabic Script-based Languages Linguistic Institute, Stanford, California, USA. July 21-22, 2007*.
12. AO'Keefe, T., Koprinska, I.: Feature Selection and Weighting Methods in Sentiment Analysis. In: *Proceedings of the 14th Australasian Document Computing Symposium (2009)*
13. Wang, S., Li, D., Wei, Y., Li, H.: A feature selection method based on Fisher's discriminant ratio for text sentiment classification. In: Liu, W., Luo, X., Wang, F.L., Lei, J. (eds.) *WISM 2009. LNCS*, vol. 5854, pp. 88–97. Springer, Heidelberg (2009)
14. Manning, C.D., Raghvan, P., Schütze, H.: *Introduction to information retrieval*. Cambridge University Press, Cambridge (2008)
15. Tan, S., Zhang, J.: An empirical study of sentiment analysis for Chinese documents. *Expert Systems with Applications* 34, 2622–2629 (2008)

16. Nicholls, C., Song, F.: Comparison of feature selection methods for sentiment analysis. In: Farzindar, A., Kešelj, V. (eds.) Canadian AI 2010. LNCS, vol. 6085, pp. 286–289. Springer, Heidelberg (2010)
17. Hang Cui, Vibhu Mittal, Mayur Datar "Comparative Experiments on Sentiment Classification for Online Product Reviews" 34, 261–269(2009))
18. Sentiment Analysis and Subjectivity. In: mIndurkha, N., Damerau, F.J. (eds.) Handbook of Natural Language Processing, 2nd ed., pp. 627–666. Chapman & Hall (2010)
19. Wang S., Li D., Wei Y., Li H.,(2009). "A Feature Selection Method based on Fisher's Discriminant Ratio for Text Sentiment Classification", *In Proceeding WISM '09 Proceedings of the International conference on Web Information Systems and Mining*, pp 88- 97.
20. Rada Mihalcea, Carmen Banea, and Janyce Wiebe. 2007. Learning multilingual subjective language via crosslingual projections. In *ACL-2007*, volume 45, page 976, 2007.
21. Manning, John Blitzer, Mark Dredze, Fernando Pereira "Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification", *In Proc. Assoc. Computational Linguistics. ACL Press*, pp 440-447, 2007.
22. Manning, Witten, I.H., Frank, E.: Data mining: Practical Machine Learning Tools and techniques. Morgan Kaufmann, San Francisco
23. WEKA, Open Source Machine Learning Software Weka. [tp://www.cs.waikato.ac.nz/ml/weka/](http://www.cs.waikato.ac.nz/ml/weka/).
24. Agarwal, B., Mittal, N.: Categorical Probability Proportion Difference (CPPD): A Feature Selection Method for Sentiment Classification. In: Proceedings of the 2nd Workshop on Sentiment Analysis where AI Meets Psychology (SAAIP 2012), COLING 2012, Mumbai, pp. 17–26 (2012)

