# Arabic sentiment analysis of YouTube comments

**Conference Paper** · October 2017

DOI: 10.1109/AEECT.2017.8257766



**3 authors:**

Abdel-Karim Al-Tamimi
Yarmouk University


Ali Mohammd Shatnawi
Jordan University of Science and Technology


Esra'a Bani Issa
Jordan University of Science and Technology

# Arabic Sentiment Analysis of YouTube Comments

Abdel-Karim Al-Tamimi
Computer Engineering Department
Yarmouk University
Irbid, Jordan
E-mail: altamimi@yu.edu.jo

Ali Shatnawi and Esraa Bani-Issa
Computer Engineering Department
Jordan University of Science and Technology
Irbid, Jordan
E-mails: ali@just.edu.jo, embaniissa14@cit.just.edu.jo

***Abstract*—With the current level of ubiquity of social media websites, obtaining the users preferences automatically became a crucial task to assess their tendencies and behaviors online. Arabic language as one of the most spoken languages in the world and the fastest growing language on the Internet motivates us to provide reliable automated tools that can perform sentiment analysis to reveal users opinions.**

**In this paper, we present our work of Arabic comments classification based on our collected and manually annotated YouTube Arabic comments. We share our classification results utilizing the most commonly used supervised classifiers: SVM-RBF, KNN, and Bernoulli NB classifiers. Experiments were performed using both raw and language-normalized datasets. We show that SVM-RBF outperformed other classification methods with an f-measure of 88.8% using a normalized dataset with two polarities.**

***Index Terms*—Natural Language Processing, Arabic Sentiment Analysis, Opinion Mining, Classification Algorithms.**

## I. INTRODUCTION

Sentiment analysis is the field that studies and analyzes people's responses and acceptance towards an entity (e.g. blogs, products, books, videos ….) using text analysis computational and algorithms to help determine people textual reactions if they are positive, negative or neutral [1].

Using sentiment analysis, fluctuations in stock prices can be predicted [2], political election preferences can be observed [3], and even radicalization groups' interactions can be tracked among many other direct and indirect benefits [4,5].

We, as individuals, tend to consult our friends and relatives about items before buying them. Companies are always eager to gather public preferences in order to attract higher numbers of customers. This is usually done by distributing surveys on a diversified group of people. Individuals and organizations find these processes resource and time consuming. These processes can even result in biased opinions, since they are usually gathered from a small group of consumers, or from the ones who are willing to give their opinions [6].

Computer science and business intelligence researchers have noted these issues since 1958 [1,6], and tried to design the appropriate algorithms that can gather opinions about a specific entity and analyze them in order to represent these opinions in an easy and a quick manner. Sentiment analysis research field has flourished since the 2000s due to the high availability and ubiquity of web-based services, and has inspired the other fields like natural language processing and information retrieval with the use of machine learning methods [1].

Sentiment analysis has been categorized into three levels: document-level, which assumes that each document contains only one sentiment about one entity. Sentence-level, which assumes that every sentence of a document can be considered as an opinion that might hold a sentiment about one entity. The third level is the aspect-level, which argues that the two previous levels do not specify the opinion's target, and do not extract the author's opinion about the various aspects of the same entity [1].

Decreasing the time of correctly reviewing opinions is the aim of various research efforts, and it is achieved generally by summarizing the opinions, eliminating spam opinions, and measuring the helpfulness and usefulness of each opinion [1].

New trends have appeared recently in the sentiment analysis research field where the analysis of authors sentiments is not only done textually, but also is done auditorily and visually [7, 8].

Arabic sentiment analysis is considered an important research activity in the field of sentiment analysis, since Arabic is the fourth spoken language and the fastest growing language on the Internet, where the number of users grows by a rate of 6.6% yearly [9]. However, sentiment analysis for Arabic language as a Morphological Rich Language (MRL) is considered a challenging task. Each Arabic root word has many derivatives, with almost different meanings and polarities for each different regional dialect. Additionally, derivatives can sometimes have different meanings and polarities [10].

Many researchers have conducted interesting researches in the field of Arabic sentiment analysis [11]. Multiple researches based their work on extracted datasets from Twitter, and analyzed them using advanced machine language algorithms like Support Vector Machines (SVM) [12, 13], Naïve Bayes (NB), and K-Nearest Neighbor (KNN) [14, 15]. Other research efforts combined Twitter datasets with Maktoob datasets [16]. Maktoob is the first Arabic/English e-mail system, the first Arabic social website, and was hosting the largest group of Arabic speaking Internet users. Other researchers focused on constructing the largest manually annotated Arabic corpus by collecting more than 63,000 Arabic book reviews [13].

While most researchers work on Modern Standard Arabic (MSA) [17, 18], others considered the importance of including other Arabic dialects in their experiments, due to the fact that MSA is not used as a colloquial language in Arabic social websites. Some considered adding Egyptian dialect [12, 18], Jordanian dialect [14, 16], Syrian dialect [19], while others incorporated all of them in addition to Iraqi, Lebanese and Saudi dialects [20]. Analyzing Arabize language (Arabic-chat alphabet) and emoticons is also an interesting research area [14]. Most of the researchers in the Arabic sentiment analysis field confined their work to produce two possible opinion polarities: positive or negative, while others expanded it by providing a ranking system from 1 to 5 stars [13], or five categories (from excellent to horrible) [17].

YouTube is one of the best sites to extract opinions and sentiments since it's considered the second accessed social website around the world, with 1,000,000,000 unique monthly visitors [21]. Many English language sentiment analysis researchers benefited from the wide spread of YouTube and performed their analyses using YouTube publicly available data. Some researchers studied only YouTube comments' sentiments for specific video categories [22]. The authors in [23] collected around 6 million comments for all YouTube video categories, and examined the dependency between the comments' ratings and their sentiments. They also examined the degree of relevance between the comment ratings and topic categories. The authors in [24] proposed an implementation for binding the users' attitudes with their locations by studying the sentiments of the comments on special videos published from previously known locations. The work presented in [3] focused on studying the interactions between jihadi YouTube group users and their profiles

Most of the English language sentiment analysis researchers conducted their sentiment analyses on their corpuses using the SentiWordnet lexicon [25] (since it contains most of the English sentiment words with three distinctive weights: positive, negative and neutral for each word). Olga Uryupina et al. [26] presented manually annotated English and Italian corpuses, and a study of the relevance between the video's comments and the video contents. To confine their approach, they collected their video sets targeting special brands of automobiles, tablets and digital cameras.

Although YouTube videos are viewed daily by 310 million users in the Arabic-spoken countries, there is a very limited number of Arabic sentiment analyses done using YouTube data [27]. To the best of our knowledge, only three papers were published on Arabic sentiment analyses that are based on YouTube data. The authors in [28] collected 4050 English and Arabic reviews/comments about Academic, News and commercial topics from Twitter, Facebook and YouTube in order to measure the accuracy of the English and Arabic sentiment lexicons by applying K-Nearest Neighbor (KNN), Naïve Bayes (NB) and Support Vector Machines (SVM) clustering methods.

Authors of [29] gathered Jordanian reviews about hotels from the same social websites used in [28]. They used SVM, Back-Propagation Neural Networks (BPNN), NB, and decision tree classifiers on their dataset after applying preprocessing techniques such as part-of-speech (POS) tagging, light stemming, root stemming, and removing stop words, punctuations and digits. They experimented with five different numbers of training-sets sizes using these classifiers, and found that when increasing the number of training reviews, the accuracy will be better except for BPNN. The best achieved f-measure of 96.06% resulted from the SVM classifier.

With more focus on video comments, the authors in [30] collected 20,000 YouTube comments on movies. They applied several preprocessing steps on the collected comments dataset including stop words removal and stemming, and then they used Samhaa lexicon [31] to get the polarity of these comments. After that, they used NB, SVM and decision tree classifiers to measure the accuracy of their methods. Their results showed 3-4% accuracy improvements over the previous works.

In this paper, we present our work of analyzing the Arabic comments of YouTube top regional videos. We provide a manually annotated YouTube comments dataset gathered from all YouTube videos' categories. Additionally, we share our finding from performing several supervised classification experiments using SVM, KNN, and NP classifiers applied to both our raw dataset and our preprocessed dataset to note the importance of the applied preprocessing steps.

The following sections of this paper contains the following: section II discusses the proposed methodology. Section III explains how we performed our analyses using our developed tools, and the achieved results. In the last section, section IV, we discuss our main observations and findings.

## II. Methodology

In this section, we explain the steps performed to assure the quality of our dataset and the applied research methodology. This research methodology main stages are: gathering the Arabic YouTube comments, preprocessing steps, comparing several supervised machine-learning classification methods, and presenting their results.

### *A. Preparing YouTube Comments Corpus*

YouTube comments did not get their fair share of analyses amongst Arabic sentiment analysis researchers due to the excessive use of dialects and other unstandardized Arabic words. We focus our research on tackling this issue by gathering a large amount of comments written about the most watched videos in the 23 countries of the Arab world. Before conducing any Arabic language preprocessing steps, we removed all non-Arabic comments, and then we chose a random sample from the remaining Arabic comments with a total of 8,053 comments.

### B. Arabic Comments Preprocessing Steps

Several preprocessing steps were applied on our random sample of comments, as explained below, to achieve a higher accuracy in the planned supervised clustering experiments as recommended in [32].

#### 1) Text Tokenization

Some Arabic words does not show the author's sentiment individually (e.g. as the words: "your voice: صوتك", "loud: مرتفع"), but when they are used in consecutively they may express a specific sentiment.

Moreover, there are some language tools in the Arabic language that are used as negation tools to flip the polarity of the comment's sentiment. However, in some cases, they work only as conjunctions depending on the context and their positions in the sentence. We used bigrams in our experiments in order to reduce the effect for these two problems. Bigrams work by splitting sentences into pairs of words, by taking each word with the preceding word once and another time with the following word [33]. We partitioned each comment into bigrams and converted them into numeric vectors by applying term frequency-inverse document frequency (tf-idf) feature vector [36].

#### 2) Normalization

Removing diacritics, normalizing similar Arabic letters shapes, and omitting duplicated characters, are the normalization steps that we applied before performing our experiments for the following reasons.

*a) Diacritics Removal*

Diacritics are used in MSA language and written above the word's letters to change the pronunciation and sometimes the meaning of the words. However, in the used dialects in social media websites, diacritics are rarely used, and in most of the cases they are used for decorations reasons only.

*b) Similar Arabic Letters Normalization*

We usually tend to write similar Arabic letters alternatively, especially the Hamza (ء) when it exists with the Aleph (ا) letter, which pronounced differently depending on its position. Generally, we are not concerned about its correct spelling since we know that the text context shows its intended meaning (e.g. we write الى instead of إلى).

Additionally, The *Ta marbuta* (ـة), which appears only at the end of words and has sort of a similar shape to the *Ha* letter (ـه ) at the end of the word too, web users often write the *Ha* and the *Ta marbuta* interchangeably. Therefore, we converted all the shapes of Aleph with the Hamza or the Madda (آ) to a normal Aleph, and all *Ta marbuta* to *Ha* letter.

*c) Duplicate Letters Removal*

Social websites' users write sometimes the same letters many times consecutively to emphasize the words meaning, or by a mistake. A simple solution to overcome this issue is to remove all the similar redundant letters. Punctuations (such as commas and dots) and repeated spaces are also to be compacted and replaced by one space letter.

Increasing the weight of the sentiment when it is emphasized is not an objective of this research. Instead, we worked with the polarities of the sentiments in a binary model representing positive sentiments as "1", negative sentiments as "-1" and neutral ones as "0".

#### 3) Filtering Repeated Comments

After applying all these normalization steps, all comments were processed by removing duplicate comments to ensure the uniqueness of the dataset in the next stages.

### C. Supervised Classification Experiments

We performed several supervised classification experiments using the most commonly used algorithms: SVM [34], KNN [35], and NB [33]. Tf-idf feature vector [36] was used to represent the linguistic features of the comments.

The supervised clustering experiments were conducted on different versions of the collected dataset: raw and normalized datasets, balanced and unbalanced datasets, datasets with three sentiments (positive, negative and neutral) and two sentiments (positive and negative), and on the entire dataset of related and unrelated comments compared to the dataset with only the topic-related comments.

## III. Implementation And Results

In this section, we explain how we gathered and annotated the comments, the used tools for each of the previous steps, and the results achieved after performing the supervised classification tests.

### A. Collecting YouTube Comments Corpus

We used Python 2.7 with YouTube API to download the comments of the top 200 viewed videos in the 23 Arab regions. We obtained 4,093,384 comments, then we used *langdetect* [36] python package to filter the non-Arabic comments, which resulted in 794,225 Arabic comments.

To manually annotate the comments to provide the ground truth to our later classification experiments, we developed a windows application tool using WPF (Windows Presentation Foundation) toolkit and C#. The tool helps annotators classify Arabic comments based on their polarities (positive, negative or neutral) and their relevance to the video (i.e. the relevance between the comments and the video's title, description, and tags).

The following rules were used to guide the annotators during this process: any comment that agrees on the video content or supports the video uploader is considered a positive-related comment. Any comment that opposes the video content or the video uploader is considered as a negative-related comment. Any comment that contains contrasting opinions (both positive and negative) about the video content or the video uploader is considered as a neutral-related comment. In addition to that, we treated spam or advertisement comments as neutral-unrelated comments.

Each comment was annotated two times by volunteer graduate students, and a third-time time by one of the authors. Only the comments with three similar annotations were considered in the final training stage with a total of 6,440 comments.

Figure 1 shows our Polarity Manual Classifier (PMC) tool that we developed and used for manually annotating the comments. We tried as much as possible to show all the needed textual information for the annotation process, such as the video's title, description, tags, views count, likes count, and dislikes count next to the comment itself.

The annotators were asked to read all this information, then decide if the comment is positive, by pressing on the green face, or negative, by pressing on the red face, or otherwise he/she should press the yellow face, indicating that it is a neutral comment. Then, the annotator is directed to choose, depending on the degree of relevance between the comment and the video's information, if the comment is related by pressing the related (مرتبط) or the unrelated (غير مرتبط) buttons that appear only after choosing the polarity.

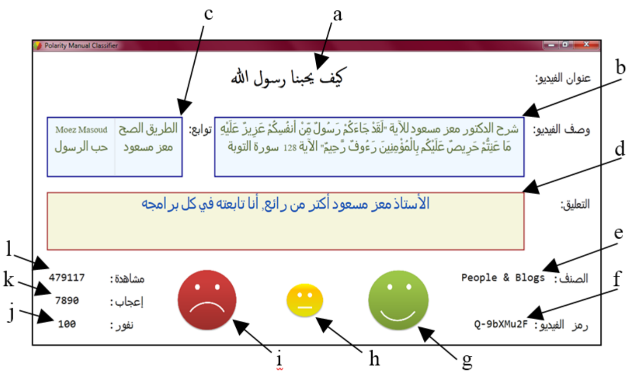


Fig. 1. Polarity Manual Classifier Tool. a) Video Title, b) Video Description, c) Video Tags, d) Video Comment, e) Video Category, f) Video ID, g) Positive Polarity Button, h) Neutral Polarity Button, i) Negative Polarity Button, j) Dislikes Count, k) Likes Count, l) Views Count.

## *B. Preprocessing Steps*

We have performed the discussed preprocessing steps din Section II using Python 2.7. For the text tokenization process, we entered the unigrams and the bigrams of each comment to the tf-idf feature vector to form the input vector to the supervised classification experiments. Some Arabic words contain duplicate letters, and since we only care about the sentimental words, we did not remove any of the duplicate letters, except in the cases where we did not find any match using Samhaa [31] sentimental lexicon.

After the entire dataset was processed using these preprocessing steps, we removed all the occurrences of any repeated comments that might have been the result of the earlier preprocessing steps. As a result, our final comments dataset contains unbalanced 5,986 comments: 4,132 positive comments, 780 neutral comments, and 1,074 negative comments.

## *C. Supervised Classification Experiments*

Python 2.7 Scikit-learn package was used to perform the supervised classification training and testing steps. We experimented using several supervised classification methods: SVM with RBF (Radial Bases function) kernel, KNN with K=1 to K=20, and Bernoulli NB classifiers. Ten folds cross validations were applied to better represent the training and testing results. These classification methods were applied on the different versions of our final dataset as discussed in Section II. Table 1 shows the experiment results of using all the classifiers: SVM-RBF, KNN and Bernoulli NB. As can be noted, the highest precision, recall and f-measure values were achieved using SVM-RBF when applied to our normalized 2-classes (positive and negative) unbalanced dataset.

SVM and KNN performed the best (in terms of precision, recall and f-measure values) when applied to the same normalized 2-classes (positive and negative) unbalanced dataset. While Bernoulli NB performed its best results with the normalized unbalanced 2-classes with only related comments.

Generally, using two classes (positive and negative) only leads to better results. We argue that the reason behind this is because we annotated all comments with contrasting opinions as a neutral, which created a difficult task for our classifiers.

Using balanced datasets resulted in inferior results to the unbalanced datasets, since they contain smaller numbers of comments (2,148 for the balanced 2-class dataset and 2,340 for the balanced 3-class dataset). The same reason applies when using the entire dataset that shows better results than when using only the related comments.

The reason that the normalized dataset achieves better results than the raw dataset is that more matches can be found between comments' words after applying the normalization rules. Positive comments have higher accuracies in the unbalanced dataset experiments, since the number of positive comments is much higher than the number of negative and neutral comments. The negative comments show the best results in the balanced datasets experiments this is due to fact that negative comments tend to contain more words than positive comments, which improves the probability of finding matches between different negative comments.

TABLE I. SUPERVISED CLASSIFCATION RESULTS

| Dataset | No. of Classes | Class | SVM | | | KNN | | | Bernoulli NB | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Precision | Recall | F-measure | Precision | Recall | F-measure | Precision | Recall | F-measure |
| Raw Dataset | Balanced 2 Classes | Negative | .835 | .71 | .767 | .607 | .91 | .728 | .91 | .438 | .589 |
| | | Positive | .72 | .84 | .778 | .82 | .41 | .546 | .63 | .957 | .76 |
| | | Average | .77 | .77 | .77 | .66 | .66 | .66 | .698 | .698 | .698 |
| | Balanced 3 Classes | Negative | .65 | .599 | .62 | .509 | .568 | .53 | .827 | .376 | .516 |
| | | Neutral | .70 | .63 | .66 | .46 | .589 | .509 | .64 | .577 | .605 |
| | | Positive | .61 | .726 | .66 | .665 | .38 | .485 | .51 | .84 | .639 |
| | | Average | .65 | .65 | .65 | .51 | .51 | .51 | .599 | .599 | .599 |
| | Unbalanced 2 Classes | Negative | .74 | .58 | .65 | .28 | .857 | .42 | .81 | .28 | .416 |
| | | Positive | .90 | .94 | .92 | .92 | .439 | .585 | .845 | .98 | .90 |
| | | Average | .875 | .875 | .875 | .52 | .52 | .52 | .84 | .84 | .84 |
| | Unbalanced 3 Classes | Negative | .69 | .53 | .60 | .23 | .86 | .37 | .778 | 27 | .40 |
| | | Neutral | .579 | .538 | .55 | .52 | .146 | .22 | .59 | .41 | .48 |
| | | Positive | .849 | .907 | .876 | .865 | .38 | .517 | .78 | .95 | .859 |
| | | Average | .79 | .79 | .79 | .438 | .438 | .438 | .765 | .765 | .765 |
| Related Raw Dataset | Balanced 2 Classes | Negative | .80 | .70 | .747 | .61 | .916 | .735 | .90 | .446 | .59 |
| | | Positive | .735 | .828 | .778 | .836 | .42 | .56 | .63 | .95 | .76 |
| | | Average | .76 | .76 | .76 | .669 | .669 | .669 | .70 | .70 | .70 |
| | Balanced 3 Classes | Negative | .58 | .457 | .51 | .455 | .587 | .50 | .815 | .288 | .42 |
| | | Neutral | .63 | .60 | .61 | .43 | .44 | .426 | .585 | .54 | .56 |
| | | Positive | .556 | .699 | .61 | .62 | .40 | .486 | .48 | .828 | .61 |
| | | Average | .586 | .586 | .586 | .478 | .478 | .478 | .55 | .55 | .55 |
| | Unbalanced 2 Classes | Negative | .81 | .57 | .67 | .269 | .78 | .40 | .82 | .28 | .418 |
| | | Positive | .90 | .966 | .93 | .90 | .49 | .635 | .85 | .98 | .91 |
| | | Average | .88 | .88 | .88 | .547 | .547 | .547 | .849 | .849 | .849 |
| | Unbalanced 3 Classes | Negative | .70 | .525 | .598 | .23 | .82 | .36 | .785 | .27 | .40 |
| | | Neutral | .42 | .36 | .387 | .35 | .07 | .11 | .396 | .23 | .29 |
| | | Positive | .86 | .92 | .89 | .878 | .40 | .545 | .81 | .965 | .88 |
| | | Average | .81 | .81 | .81 | .455 | .455 | .455 | .79 | .79 | .79 |
| Normalized Dataset | Balanced 2 Classes | Negative | .86 | .72 | .78 | .639 | .909 | .75 | .91 | .508 | .65 |
| | | Positive | .76 | .88 | .81 | .845 | .485 | .61 | .66 | .95 | .779 |
| | | Average | .80 | .80 | .80 | .697 | .697 | .697 | .73 | .73 | .73 |
| | Balanced 3 Classes | Negative | .715 | .577 | .636 | .448 | .82 | .579 | .83 | .396 | .53 |
| | | Neutral | .676 | .699 | .685 | .647 | .35 | .45 | .67 | .64 | .65 |
| | | Positive | .667 | .76 | .71 | .706 | .43 | .53 | .548 | .855 | .667 |
| | | Average | .68 | .68 | .68 | .536 | .536 | .536 | .63 | .63 | .63 |
| | Unbalanced 2 Classes | Negative | .78 | .63 | .70 | .569 | .315 | .36 | .885 | .365 | .516 |
| | | Positive | .90 | .95 | .93 | .84 | .90 | .859 | .856 | .987 | .917 |
| | | Average | **.888** | **.888** | **.888** | **.78** | **.78** | **.78** | .859 | .859 | .859 |
| | Unbalanced 3 Classes | Negative | .72 | .588 | .649 | .50 | .30 | .33 | .84 | .34 | .486 |
| | | Neutral | .59 | .555 | .57 | .586 | .228 | .32 | .596 | .465 | .51 |
| | | Positive | .86 | .91 | .887 | .76 | .88 | .80 | .799 | .95 | .87 |
| | | Average | .809 | .809 | .809 | .69 | .69 | .69 | .78 | .78 | .78 |
| Related Normalized Dataset | Balanced 2 Classes | Negative | .866 | .717 | .785 | .63 | .898 | .74 | .92 | .49 | .64 |
| | | Positive | .76 | .89 | .819 | .82 | .478 | .60 | .655 | .957 | .777 |
| | | Average | .80 | .80 | .80 | .688 | .688 | .688 | .725 | .725 | .725 |
| | Balanced 3 Classes | Negative | .70 | .459 | .55 | .44 | .84 | .58 | .827 | .24 | .36 |
| | | Neutral | .62 | .65 | .637 | .647 | .299 | .40 | .59 | .57 | .58 |
| | | Positive | .61 | .797 | .695 | .73 | .44 | .55 | .507 | .877 | .64 |
| | | Average | .636 | .636 | .636 | .53 | .53 | .53 | .56 | .56 | .56 |
| | Unbalanced 2 Classes | Negative | .81 | .57 | .67 | .56 | .31 | .36 | .87 | .34 | .489 |
| | | Positive | .90 | .966 | .93 | .84 | .91 | .869 | .858 | .987 | .91 |
| | | Average | .88 | .88 | .88 | .79 | .79 | .79 | **.8597** | **.8597** | **.8597** |
| | Unbalanced 3 Classes | Negative | .75 | .579 | .65 | .52 | .30 | .346 | .84 | .32 | .46 |
| | | Neutral | .44 | .379 | .406 | .47 | .155 | .23 | .41 | .257 | .31 |
| | | Positive | .87 | .93 | .90 | .797 | .90 | .837 | .817 | .97 | .887 |
| | | Average | .83 | .83 | .83 | .738 | .738 | .738 | .80 | .80 | .80 |

## IV. CONCLUSIONS

In this research, we presented our Arabic sentiment analysis work using 5,986 manually annotated Arabic YouTube comments gathered from the top viewed videos in the Arabic world. We performed and compared several supervised classification experiments using SVM-RBF, Bernoulli NB and KNN classifiers.

Based on our experiments results, SVM-RBF achieved the highest f-measure of 88.8% when using our unbalanced 2-classes (positive and negative) normalized dataset that contained both related and unrelated comments. We observed that the preprocessing and normalization steps always improved the classification outcomes especially when using larger datasets.

## REFERENCES


[1] B. Pang and L. Lee, "Sentiment Analysis and Opinion Mining" Foundations and Trends in Information Retrieval, 2(1-2), pp. 1135, 2008.

[2] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment," in the 4th International Conference on Weblogs and Social Media (ICWSM-2010), 2010.

[3] A. Bermingham, M. Conway, L. McInerney, N. O'Hare, and A. Smeaton, "Combining Social Network Analysis and Sentiment Analysis to Explore the Potential for Online Radicalisation," in the International Conference on Advances in Social Network Analysis and Mining (ASONAM '09), pp. 231-236, 2009.

[4] J. Karlgren, M. Sahlgren, F. Olsson, F. Espinoza, and O. Hamfors"Usefulness of Sentiment Analysis ," in the 34th European conference on Advances in Information Retrieval, 2012.

[5] B. Liu, "Sentiment Analysis and Opinion Mining," Morgan and Claypool, 2012.

[6] H. P. Luhn, "A Business Intelligence System," IBM Journal of Research and Development, 1958.

[7] S. Poria, E. Cambria, N. Howard, G. Huang and A. Hussain, "Fusing audio, visual and textual clues for sentiment analysis from multimodal content," in Neurocomputing, 174, pp. 50-59, 2016, Elsevier.

[8] M. Wollmer, F. Weninger, T. Knaup, B. Schuller, S. Congkai, K. Sagae and L. Morency, "Youtube movie reviews: Sentiment analysis in an audio-visual context," in IEEE Intell. Syst., vol. 28, no. 3, pp. 46-53, 2013

[9] Internetworldstats.com , "Internet World Users by Language," 2016. [Online] Available: http://www.internetworldstats.com/stats7.html.

[10] M. Abdul-Mageed, M. Diab, and S. Kübler, "SAMAR: Subjectivity and sentiment analysis for Arabic social media," Computer Speech & Language, vol. 28, no. 1, pp. 20-37, 2014.

[11] M. Biltawi, W. Etaiwi, S. Tedmori, A. Hudaib, and A. Awajan, "Sentiment classification techniques for Arabic language: A survey," Information and Communication Systems (ICICS), 2016 7th International Conference on. IEEE, 2016.

[12] A. Shoukry and A. Rafea, "Sentence-level Arabic sentiment analysis," in 2012 International Conference on Collaboration Technologies and Systems (CTS), IEEE, 2012.

[13] M. Nabil, M. Aly, and A. Atiya, "LABR: A Large Scale Arabic Sentiment Analysis Benchmark," 2015.

[14] R. Duwairi, R. Marji, N. Sha'ban, and S. Rushaidat, "Sentiment Analysis in Arabic tweets," in 5th International Conference on Information and Communication Systems (ICICS), 2014.

[15] R. Duwairi, "Arabic Sentiment Analysis Using Supervised Classification," in the IEEE 2nd International Conference on Future Internet of Things and Cloud (FiCloud-2014), Spain, 2014.

[16] N. A. Abdulla, N. A. Ahmed, M. A. Shehab, M. Al-Ayyoub, M. N. Al-Kabi, and S. Al-rifai, "Towards improving the lexicon-based approach for Arabic sentiment analysis," International Journal of Information Technology and Web Engineering, vol. 9, no. 3, pp. 55–71, 2014.

[17] W. Cherif, A. Madani, and M. Kissi, "Towards an Efficient Opinion Measurement in Arabic Comments," Procedia Computer Science, vol. 73, pp. 122–129, 2015.

[18] H. Ibrahim, S. Abdou, and M. Gheith, "Sentiment Analysis for Modern Standard Arabic and Colloquial," IJNLC International Journal on Natural Language Computing, vol. 4, no. 2, pp. 95–109, 2015.

[19] S. Mohammad, M. Salameh, and S. Kiritchenko, "How Translation Alters Sentiment," Journal of Artificial Intelligence Research, vol. 55, pp. 95–130, 2016.

[20] M. Al-Kabi, A. Gigieh, I. Alsmadi, H. Wahsheh, and M. Haidar, "An Opinion Analysis Tool for Colloquial and Standard Arabic, " in the 4th International Conference on Information and Communication Systems (ICICS 2013), 2013.

[21] Ebimba.com, "Top 15 Most Popular social Networking Sites," 2016. [online]. Available: http://www.ebizmba.com/articles/social-networking-websites . [Accessed: 17- Aug- 2016]

[22] S. Choudhury, and J. G. Breslin. "User sentiment detection: a YouTube use case," in the 21st National Conference on Artificial Intelligence and Cognitive Science, 2010.

[23] S. Siersdorfer, S. Chelaru, W. Nejdl, and J. San Pedro, "How useful are your comments?: analyzing and predicting youtube comments and comment ratings, " in the 19th international conference on World wide web, pp. 891-900, 2010, ACM.

[24] E. Polymerou, D. Chatzakou and A. Vakali, "Emotube: A sentiment analysis integrated environment for social web content," in the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS'14), 2014.

[25] SentiWordNet, [online]. Available: http://sentiwordnet.isti.cnr.it/ . [Accessed: 17- Aug- 2016]

[26] O. Uryupina, B. Plank, A. Moschitti, A. Rotondi, and A. Severyn, "SenTube: A corpus for sentiment analysis on YouTube social media," in LREC, 2014.

[27] Albawaba.com, "YouTube at 10 years old: MENA ranks second worldwide for video views," 2015. [online]. Available: http://www.albawaba.com/business/youtube-10-years-old-mena-ranks-second-worldwide-video-views-711134 . [Accessed: 17- Aug- 2016]

[28] M. Al-Kabi, N. M. Al-Qudah, I. Alsmadi, M. Dabour, and H. Wahsheh, "Arabic / English Sentiment Analysis: An Empirical Study, " in the 4th International Conference on Information and Communication Systems (ICICS 2013), 2013.

[29] M. Hammad and M. Al-awadi, "Sentiment Analysis for Arabic Reviews in Social Networks Using Machine Learning," In Information Technology: New Generations, pp. 131-139, Springer International Publishing, 2016.

[30] R. Elawady, S. Barakat and N. Elrashidy, "Sentiment Analyzer for Arabic Comments," International Journal of Information Science and Intelligent System, vol. 3, no. 4, pp. 73-86, 2014.

[31] S. El-Beltagy, and A. Ali, "Open issues in the sentiment analysis of Arabic social media: A case study," Innovations in information technology (iit), 2013 9th international conference on, pp. 215-220, IEEE, 2013.

[32] A. Ayedh, G. Tan, K. Alwesabi, and H. Rajeh, "The effect of preprocessing on arabic document categorization," Algorithms, 2016.

[33] C. Manning, P. Raghavan, and H. Schütze, "Introduction to information retrieval," Cambridge: Cambridge university press, 2008.

[34] C. Cortes and V. Vapnik, "Support-vector networks," Machine learning, vol. 20, no. 3, pp.273-97,1995.

[35] N. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," The American Statistician, vol. 46, no. 3, pp. 175-185, 1992.

[36] J. Leskovec, A. Rajaraman, and J. Ullman, "Mining of massive datasets," Cambridge University Press, 2014, ch. 1, sec. 3, pp. 8-9.

[37] M. Danilak, langdetect: language detection library, 2014, [online]. Available: https://pypi.python.org/pypi/langdetect . [Accessed: 14- May-2017].