ISSN 2347 - 3983
Volume 8. No. 2, February 2020
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter46822020.pdf
https://doi.org/10.30534/ijeter/2020/46822020

Opinion Mining: How efficient are Online classification tools?

Hayel khafajeh [1]
[1]Zarqa University, Jordan, HayelKH@zu.edu.jo

ABSTRACT

Recently, Online Social Networks (OSNs) are considered as important resource of information, since they provide a huge amount of data that reflects the interactions between users in various fields, such as: politics, sport and business. Opinion mining (or sentiment analysis) is a process that uses natural language processing, and text analysis methods to understand users' feelings or opinions, and detect their polarity, which could be positive, negative or neutral. The outcomes of opinion mining approaches help in extracting useful patterns that enable traders to take critical decisions for business, marketing and politics. In the literature, we have several proposed opinion mining systems, tools and approaches, but in general they are not available on public. Many other online opinion mining tools are simple to use and available for free or as demos. Opinion mining online tools performance need to be evaluated to attract researchers and companies utilizing their advantages. The main purpose of this study is to evaluate how efficient are online opinion mining tools for Arabic language. We used benchmark Arabic opinion collections and classify them using two popular online sentiment analysis tools that support Arabic language; Paralleldots and Repustate. The experiment used prediction quality measurement to evaluate these tools and compare their results with several machine learning classifiers in order to recommend the best available solution for Arabic sentiment analysis. Our results showed that Paralleldots API is highly recommended for Arabic sentiment analysis for both positive and negative reviews.

Key words : Online Social Networks, Sentiment Analysis, Polarity, Paralleldots, Repustate.

1. INTRODUCTION

Individuals more and more be likely to distribute online reviews and comments about several topics or various goods using blogs and social media. Online Social Networks (OSNs) users reviews have powerfully impact on other user's choices, more than paid advertising to buy goods, select a movie to watch, visit a particular location or to participate in a political event [1]. The huge volume of existing data needs an automatic analysis to obtain beneficial outcomes in in decision-making. Automatic analysis includes fetching the useful information depending on valuable features from the user's reviews that can help, i.e. to highlight the reputation of movies from online comments and deciding which model of a smartphone camera has the greatest capabilities. In addition, automatic analysis can help customers to know the benefits of the mobile applications before install them or recommending a wonderful restaurant depending on the previous user's opinions [1]. Automatic analysis can enhance the marketing field by tracking user's comments, meet their requirements and produce improved quality services. Textual opinions have been significantly growing in latest years, because of the widespread of marketing, online learning and social communication. Beside the users textual opinions there are increasing number of audio and video contributing opinions on social networks, i.e. Twitter, Facebook and YouTube that could be converted into textual content as a way to be processed [2]. Figure 1 presents the growth of Facebook users in the Arab world during 2011-2017 [3]

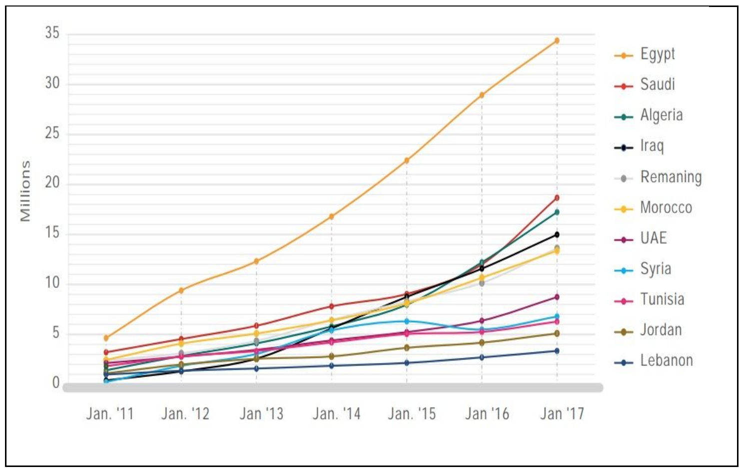

Figure 1: Growth of Facebook Users in the Arab World During 2011-2017.

Opinion mining or sentiment analysis is the process of automatically analyzing, preprocessing, extracting the useful attributes (i.e. words or phrases) and identifying the relationships among the content to classify the users opinion feeling (class) and expressions regarding particular state.

Preprocessing could include one or more of tokenization, stop word removing, normalization or stemming [4]. Opinion mining can be divided into several levels [5] as subjective or objective. Polarity classification concerns on deciding if the subjective opinions/emotion is positive or negative. Arabic Opinion mining is using to make suitable companies and organizations decision making. Figure 2 shows language use on Twitter through the Arab world [3]

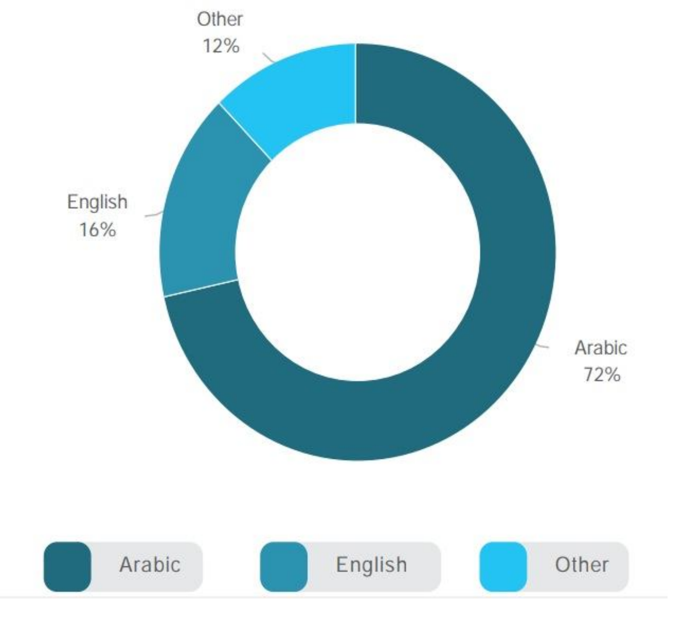

Figure 2: Growth of Facebook Users in the Arab World During 2011-2017.

There are several researches and systems on detecting Arabic opinions polarity including supervised, semi-supervised and unsupervised techniques that evaluates among various corpuses and obtained remarkable accuracy results [6, 7 and 8]. In general, the proposed models and system are not available for public use, some other opinion mining tools are simple to use and available for free as demos. The existing online tools need to evaluate their performance to attract researchers and companies to get benefit of their features. The main target of this study is to evaluate how efficient are online opinion mining tools for Arabic language; Paralleldots [9] and Repustate [10] as case studies. Our contributions can be summarized as follows: (a) we adopt benchmark free available Arabic sentiment analysis dataset [11]; (b) we use ParallelDots AI API Sentiment Analysis and Repustate Sentiment Analysis as case studies of popular, available, free and online sentiment analysis tools that support Arabic language [9] and [10]; (c) we simulate the user interaction with the ParallelDots AI API, Repustate API and classify the data collection into positive, negative or neutral polarity; (d) our experiments evaluate the ParallelDots AI API, Repustate API and seven machine learning classifiers using prediction quality measurements, and recommend the best technique for Arabic sentiment analysis. The rest of this paper is organized as it follows: Section 2 presents the related work to this study, and section 3 highlights the proposed methodology. Section 4 shows the experiments and results. Section 5 presents the conclusion and future work.

2. RELATED WORK

There are several studies focused on enhancing Arabic sentiment analysis but few of them take into consideration the evaluation of online Arabic sentiment analysis tools.

The study of [12] proposed sentiment analysis tool called SAMAR for detecting the polarity of Arabic posts in social networks. They used part of speech (POS) tags and lexemes to decide the polarity of subjectivity analysis. They highlighted the issues of Arabic dialects in sentiment analysis.

The study of [13] used in their work distant supervision (DS) methods for detecting subjectivity and polarity for Twitter posts. Their learning approach depends on large automatically labeled Arabic comments through Twitter network. The results showed that DS enhanced the effectiveness of the detecting performance for subjective posts that include emoticons.

The study of [14] researchers collected new Arabic labeled dataset that includes MSA and several Arabic dialects. The researchers applied random graph walk algorithm and the results showed improvement on the polarity detecting performance besides using POS, normalization and light stemming methods.

The study of [15] used Large Arabic Book Reviews (LABR) dataset that contains 63K reviews. The researchers selected 25% of the available reviews and applied stop word removal, normalization, tokenization and stemming techniques. The experiments evaluated several machine learning algorithms and the results showed that Multinomial Naive Bayes (MNB) is the best classifier for detecting the polarity.

In [5] authors evaluated online sentiment analysis Social-Mention (http://socialmention.com) using 4,050 Arabic opinions. They depend on polarity dictionaries i.e. Arabic and Emoticons. The results showed that Social-mention tool achieved an accuracy of 66.2%. The study of [16] provided an application of Arabic opinion mining at the sentence-level through sentiment classification of Arabic tweets. They analyzed these tweets and decided their polarity as positive or negative. The researchers employed supervised machine learning approach and used Support Vector Machine (SVM). The experiment used Egyptian dialect dataset and evaluated them using SVM and Naïve Bayes (NB).

The study of [17] evaluates SentiStrength online tool for Arabic language. The researchers used 11 Arabic datasets that belonging to several topics and dialects and the results showed an accuracy of 62%.

In [18] researchers employ ParallelDots AI API, in order to analyze Twitter messages that belong to 14th Gujarat Legislative Assembly Election, 2017. Polarity detection showed the opportunity of winning party. The experiments results showed ParallelDots AI API as strong and help tool for summarizing people sentiment for decision making.

3. METHODOLOGY

The main target of this study is to evaluate how efficient are the online opinion mining tools for Arabic language. So, the proposed methodology consists of the following phases:

1. Adopt a benchmark of free and available data collection for Arabic reviews.
2. Use ParallelDots AI API Sentiment Analysis and Repustate as case studies of popular, available, free and online sentiment analysis tools that support Arabic language.
3. Simulate the user interaction with both ParallelDots AI API, and Repustate and classify the data collection, into: positive, negative and neutral classes.
4. Apply seven machine learning classifiers on the benchmark data collection.
5. Evaluate ParallelDots and Repustate online tools using prediction quality measurements.
6. Compare ParallelDots and Repustate results with machine learning classifiers results and recommend the best solution for detecting Arabic sentiment analysis.

3.1 Arabic Sentiment Analysis Benchmark Dataset

Choosing a benchmark Arabic dataset is important, due the observation that some Arabic data collections suffer from incorrect user's polarity labeling which effect on the training dataset. In the literature, we found some public and available Arabic opinions datasets such as: [19], but we excluded it due that did not cover Arabic neutral reviews. The study of [11] provides public Arabic social network opinions that cover all polarity classes, i.e. positive, negative and neutral. The dataset of [11] was collected in 2015, from Maktoob Yahoo! social network that is diverse, flexible and relatively standard Arabic sentiment analysis dataset including opinions written in the several Arabic dialects. The total amount of Arabic topics and opinions are 250 and 1,442, respectively. The topics are divided into five fields such as: Economy, Food-Life style, Religion, Sport, and Technology. The majority of this dataset (64.081%) still refers to Modern Standard Arabic (MSA), since it is theoretical to be the most used version of Arabic across the 22 countries that used it as official language. Moreover, the dataset is characterized by including various Arabic dialects and this gives diversity to our experiments. Table 1 presents the languages distribution of the adopted dataset.

Table 1: Languages distribution of the adopted dataset

Language	Percentage
MSA	65.43%
Egyptian	18.59%
Levantine	5.70%
Arabian Peninsula	2.46%
Mesopotamian Group	2.46%
Arabizi	0.385%
Maghrebi group	0.308%
English /other languages	The remaining percentage

3.2 ParallelDots AI API Sentiment Analysis

The developer of ParallelDots claimed that they are one of the best artificial intelligent research center in the world. Their work is to deal with business challenges and to provide consulting to present what, why, how and who about applying AI in business issues [9].

ParallelDots API supported by professors in several universities and research centers that focusing in Machine Learning applications [9]. Several previous studies used Parallel Dots API for non-Arabic languages, which indicate the popularity of the API and its widespread usage [18], [20], [21], [22] and [23].

ParallelDots sentiment analysis API offers high accurate analysis of the overall polarity of the textual content from several sources such as: twitter tweets, Facebook comments, Blogs, Articles, forums etc. ParallelDots can be employed for detecting the polarity to help the customer service and enhancing marketing targets. ParallelDots uses Long Short Term Memory (LSTM) algorithms [24], which is a recurrent neural network (RNN) approach of deep learning, in order to detect the textual content polarity as positive, negative or neutral. LSTMs model utilizes social network dataset as a training corpus to test the new reviews. Figure 3 shows ParallelDots polarity result with an example of Arabic comment "أتمنى لك التوفيق والنجاح", "I wish you all the best".

Figure 3: ParallelDots Positive Polarity with an Example of Arabic Comment.

Figure 4 presents an example of neutral reviews " الاجهزة الخلوية في الاردن", "Mobile Devices in Jordan" with ParallelDots polarity detection.

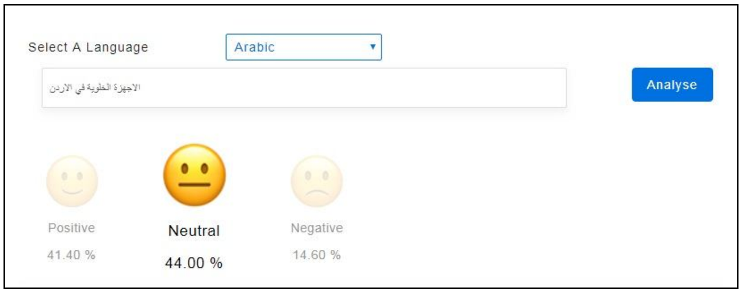

Figure 4: ParallelDots Neutral Polarity with an Example of Arabic Comment.

Figure 5 shows another example of ParallelDots polarity result with negative polarity of Arabic comment " انا لا أحب الامتحانات الطويلة", "I do not like long exams".

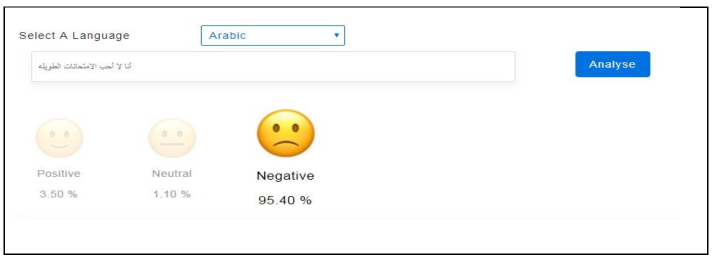

Figure 5: ParallelDots Negative Polarity with an Example of Negative Arabic Comment.

Figure 5 indicates that ParallelDots API robust against tricky comments that contain Arabic negation tools that change the polarity.

As shown in Figures 3, 4 and 5 ParallelDots computes the polarity scores for all polarity classes, then give the final result with the highest score. ParallelDots adopted the following procedures to detect the polarity for the given comments:

1. Data pre-processing to filter the content such as: remove the links and punctuations.
2. Convert every word (feature) in the comment to the corresponding vector (numeric representation).
3. Send the generated vectors to recurrent neural network (RNN) and the classification layer to detect the final polarity.
4. Final polarity is compared later with the human tagged labels to compute the error percentage to optimize the neural network results.

ParallelDots API adopts JavaScript Object Notation (JSON) which is human-readable, language-independent, very common and widely used data format. In addition, ParallelDots API supports 15 different language such as: Arabic, English and Japanese language [9].

We used the free version of ParallelDots AI API and we simulate the user interaction it to classify the dataset and generate the polarity results. The free version of this tool is limited. It took around 3 months to obtain the polarity results for our 1,255 opinions

3.3 Repustate Sentiment Analysis API

The developer of Repustate Sentiment Analysis API aimed to present it as flexible and comprehensive tool that serve customers in several applications [10].

Repustate Sentiment Analysis API employs machine learning classifiers and can be easily integrated into any workflow. It supported 24 languages and various libraries such as: C#, Java, PHP and Python. Repustate Sentiment Analysis API uses sentiment and semantic analysis that handle emoticons and emoji [10].

Since that Arabic language is unique and differs from English, Repustate Sentiment Analysis API considers that and develops several techniques to deal with it. Repustate adopts Arabic part of speech tagging which token the opinion into Verbs, nouns and adjectives and other grammatical constructs to the accurate polarity [10].

In addition Repustate Sentiment Analysis API uses many Arabic language sentiment models such as: negation various phrases, idioms and expressions that help to determine Arabic polarity. Repustate Sentiment Analysis API computes a score for the given opinion as a value between 1 and -1, in which the positive value means positive polarity, negative for the negative polarity and zero for the neutral polarity [10].

Similar to ParallelDots API, we used the free version of Repustate and we classify the dataset and obtain the polarity results. The free version of this tool is limited, so it took around 3 months to get the polarity results for our 1,255 opinions.

ParallelDots sentiment analysis API [9] and Repustate Sentiment Analysis API [10] do not detect Arabic spam i.e. unwanted or irrelevant comments [25]. In which Arabic spam detection requires special tools and techniques [25-28]. So we exclude spam opinions from our experiments and the remaining dataset size is 1,255 opinions.

4. EXPERIMENTS AND RESULTS

In this section, we will present the obtained results of the experiment in details, in order to recommend the best solution for Arabic polarity detection. We evaluated seven machine learning classifiers using 10-fold cross-validation [29]. 10-fold cross-validation divide the dataset into 10 divisions, use 9 as training dataset and one as test dataset and repeat this process 10 times to test all 10 divisions. Moreover, to evaluate ParallelDots API and Repustate API we compute the quality measurements to describe the performance of ParallelDots API and Repustate API polarity classifications. We used True Positive (TP), False Positive (FP), False Negative (FN) to calculate the Precision (P), Recall (R) and F-Measure (F-M), as shown in formulas1-3 [29].

$$Recall_i = \frac{TP}{TP + FN} \quad (1)$$

$$Precision_i = \frac{TP}{TP + FP} \quad (2)$$

$$F-measure = \frac{(2 \times TP)}{(2 \times TP) + FP + FN} \quad (3)$$

4.1 Support Vector Machines

Support Vector Machines (SVM) depends on decision planes that divides between groups of instances that containing several class labels. SVM yields better results in high dimensional space and in binary classification cases [30].
The experiments showed that the accuracy of SVM yielded 61.4343% with an error rate of 38.5657%, Table 2 shows detailed SVM results.

Table 2: Detailed SVM results

Class	Precision	Recall	F-M
Negative	0.670	0.700	0.685
Neutral	0.222	0.014	0.026
Positive	0.558	0.683	0.614
Weighted Avg.	0.575	0.614	0.582

SVM correctly classified instances of 771 out of our adopted benchmark, and 484 as incorrectly classified instances, Table 3 presents a confusion matrix of SVM.

Table 3: Confusion matrix of SVM

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	441	4	185
Neutral	68	2	75
Positive	149	3	328

As shown in Tables 2 and 3, SVM successfully yielded high acceptable detection results (more than 61%) as weighted average, and between 60-70% for detecting negative and positive classes. While SVM failed to detect neutral class.

4.2 K-Nearest-Neighbors (K-NN)

The K-Nearest-Neighbors (K-NN) is a lazy learning classifier that widely used for classification and regression purposes. K-NN learner depends on finding the distance of the test instance from each training instance and detect the output class that based on the nearest class instance of the defined k (a small positive integer value) [31].
The results of applying K-NN, when k=3 yielded an accuracy of 38.3267% and an error of 61.6733%. Table 4 presents the detailed K-NN outcomes.

Table 4: Detailed K-NN results

Class	Precision	Recall	F-M
Negative	0.500	0.006	0.013
Neutral	0.000	0.000	0.000
Positive	0.383	0.994	0.553
Weighted Avg.	0.397	0.383	0.218

K-NN failed to detect both negative and neutral classes, while it yielded high accuracy results for detecting the polarity of positive class. K-NN incorrectly classified instances of 774, while it correctly classified instances of 481. Table 5 shows a confusion matrix of K-NN.

Table 5: Confusion matrix of K-NN

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	4	1	625
Neutral	1	0	144
Positive	3	0	477

4.3 Decision Tree

Decision Tree (J48) algorithm is one of the supervised learning algorithms that capable to solve both regression and classification purposes. Decision Tree create training model

to detect the class label of the target output by using decision rules that derived from the training data features. Decision Tree visualizes these rules by creating tree representation, in which every node corresponds to a feature and every leaf node express the class label [32].
The evaluation results of Decision Tree obtained an accuracy of 52.5896% and an error rate of 47.4104%. Table 6 provides more details of decision tree results.

Table 6: Detailed Decision Tree results

Class	Precision	Recall	F-M
Negative	0.616	0.554	0.583
Neutral	0.250	0.103	0.146
Positive	0.471	0.617	0.534
Weighted Avg.	0.518	0.526	0.514

Decision Tree succeeded to find the polarity for negative and positive classes with acceptable accurate percentages, while it failed to detect neutral class. Decision Tree correctly predicted instances of 660, while it incorrectly predicted 595 instances. Table 7 shows a confusion matrix of Decision Tree.

Table 7: Confusion matrix of Decision Tree

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	349	30	251
Neutral	49	15	81
Positive	169	15	296

As shown in Tables 2 and 3, SVM successfully yielded high acceptable detection results (more than 61%) as weighted average, and between 60-70% for detecting negative and positive classes. While SVM failed to detect neutral class.

4.4 Random Forest

Random Forest (RF) is an ensemble learning classifier that used for classification and regression problems. It create multitude decision trees as training model with considering to dataset overfitting [33].

Applying Random Forest achieved accuracy results of 60.9562% and error of 39.0438%. Detailed results are shown in Table 8.

Table 8: Detailed Random Forest results

Class	Precision	Recall	F-M
Negative	0.661	0.768	0.711
Neutral	0.250	0.007	0.013
Positive	0.539	0.583	0.561
Weighted Avg.	0.567	0.610	0.573

Random Forest succeeded to detect the polarity for negative class, obtained acceptable detection rate for positive class, and failed to find the neutral class. The correctly detected instances were 765 with 490 as incorrectly classified. Table 9 explores a confusion matrix of Random Forest.

Table 9: Confusion matrix of Random Forest

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	484	3	143
Neutral	48	1	96
Positive	200	0	280

4.5 Bagging Classifier

Bagging is an ensemble method that combines the predictions from multiple machine learning algorithms together to enhance the accurate predictions instead of an individual one. Bagging used to decrease the variance of some algorithm with high variance like Decision Tree [34].
Bagging evaluation results yielded an accuracy of 56.0159% and error of 43.9841, Table 10 shows the detailed outcomes of Bagging classifier.

Table 10: Detailed Bagging results

Class	Precision	Recall	F-M
Negative	0.618	0.692	0.653
Neutral	0.179	0.034	0.058
Positive	0.503	0.546	0.523
Weighted Avg.	0.523	0.560	0.535

Similar to the previous algorithms Bagging classier failed to detect the neutral class, while it capable to detect negative and positive with acceptable percentages. The correctly classified instances were 703 with 552 as incorrectly predicted instances. Table 11 explores a confusion matrix of Bagging classifier.

Table 9: Confusion matrix of Bagging classifier

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	436	16	178
Neutral	59	5	81
Positive	211	7	262

4.6 Decision Table

Decision Table (DT) is a classifier that composed of hierarchical table, in which every node in the higher level has sub-nodes of a pair of additional features from another table. Visualization approach of this classifier make it easy to understand and more useful even with unaware machine learning experience [35].

Applying Decision Table obtained an accuracy of 52.9084% and error rate of 47.0916%. The detailed quality measurements are shown in Table 12.

Table 12: Detailed Decision Table results

Class	Precision	Recall	F-M
Negative	0.517	0.968	0.674
Neutral	0.000	0.000	0.000
Positive	0.783	0.113	0.197
Weighted Avg.	0.559	0.529	0.414

Decision Table obtained the acceptable results for detecting the polarity of negative class, while it failed to detect both neutral and positive classes. Decision Table confusion matrix is shown in Table 13.

Table 13: Confusion matrix of Decision Table

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	610	6	14
Neutral	144	0	1
Positive	425	1	54

Confusion matrix of Decision Table presented that the correctly classified instances were 664 and the incorrectly detected instances were 591.

4.7 Naive Bayes

Naive Bayes (NB) classifier is a highly scalable probabilistic classifier that depends on bayes' theorem with strong independence assumptions between the attributes. NB considers the value of specific attribute as independent of any value of other attributes. NB use textual features of textual materials to find the closeness to each predetermined labels [36] and [37].

Naive Bayes (NB) results showed an accuracy of 40.0797% and error of 59.9203%. Quality measurements details are shown in Table 14.

Table 14: Detailed NB results

Class	Precision	Recall	F-M
Negative	0.657	0.465	0.545
Neutral	0.123	0.414	0.189
Positive	0.469	0.313	0.375
Weighted Avg.	0.523	0.401	0.439

Although NB results did not reach the acceptable percentage (less than 50%), but NB yielded closest accuracy detection for all the classes. NB confusion matrix is shown in Table 15.

Table 15: Confusion matrix of NB classifier

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	293	204	133
Neutral	48	60	37
Positive	105	225	150

Confusion matrix of NB showed that the correctly predicted instances were 503 and the incorrectly detected instances were 752.

4.8 Repustate Sentiment Analysis API

Repustate Sentiment Analysis API results showed an accuracy of 47.012% and error of 52.988%. The detailed results of prediction measurements are shown in Table 16.

Table 16: Detailed Repustate API results

Class	Precision	Recall	F-M
Negative	0.664	0.465	0.546
Neutral	0.164	0.335	0.220
Positive	0.478	0.517	0.478
Weighted Avg.	0.435	0.439	0.414

The results of Repustate Sentiment Analysis API obtained acceptable results for detecting the polarity of positive class, while it failed to detect both neutral and negative classes. Repustate confusion matrix is shown in Table 17.

Table 17: Confusion matrix of Repustate API

Actual Class	Predicted Class		
	Negative	Neutral	Positive
Negative	293	204	133
Neutral	48	60	37
Positive	105	225	150

Confusion matrix of Repustate API showed that the correctly predicted instances were 590 and the incorrectly detected instances were 665.

4.9 ParallelDots API

ParallelDots API results showed an accuracy of 62.789% and error of 37.211%. The detailed results of prediction measurements are shown in Table 18.

Table 18: Detailed ParallelDots API results

Class	Precision	Recall	F-M
Negative	0.8059	0.52063	0.6325
Neutral	0.28934	0.3931	0.3333
Positive	0.61905	0.83958	0.7126
Weighted Avg.	0.57143	0.5844	0.5559

ParallelDots API results yielded high detection results for positive class, and acceptable level for negative class. Although that ParallelDots API results did not obtain the acceptable level for detecting the polarity of neutral class, but the results still better than previous (4.1-4.7) machine learning algorithms and Repustate API results, where all previous results except ParallelDots failed to detect the polarity of neutral class. Table 19 shows the Confusion matrix of ParallelDots API.

Table 19: Confusion matrix of ParallelDots API

Actual Class	Predicted Class		
	Negative	**Neutral**	**Positive**
Negative	328	112	190
Neutral	30	57	58
Positive	49	28	403

Confusion matrix of ParallelDots API showed that the correctly predicted instances were 788 and the incorrectly detected instances were 467. Figure 6 presents the comparisons of the all solutions for Arabic sentiment Analysis.

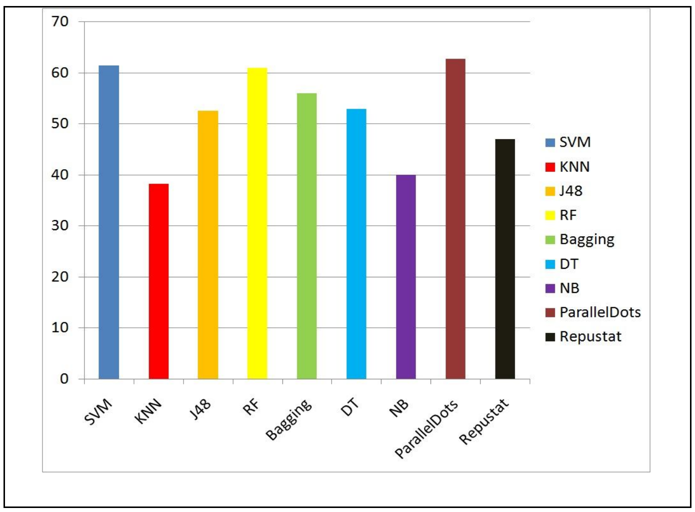

Figure 6: Comparisons of the accuracy for Arabic Sentiment Analysis Classifiers.

The overall accuracy results of K-NN, Naive Bayes and Repustate are enough to indicate that these solutions are not suitable for detecting the Arabic sentiment analysis depending on our benchmark dataset. For detecting negative and positive polarity we can use SVM, Random Forest and bagging classifiers but they are limited with neutral class detection. Depending on our obtained outcomes we are highly recommending to use Paralleldots API for Arabic sentiment analysis for both positive and negative reviews, since that Paralleldots API yielded the best results among all other solutions. Although the polarity detection accuracy of the neutral class required enhancements to reach the acceptable level (i.e. 50%), the current result of Paralleldots API is still make sense compared to all solutions that used in this study.

5. CONCLUSION

Sentiment analysis aims to understand, analyze, and extract useful features from OSNs user's feedback to meet their needs and produce enhanced quality services [38]. Several Arabic sentiment analysis researches are available with different adopted approaches but usually they are not available on public. The main target of this study is to evaluate how efficient are online opinion mining tools for Arabic language; we used Paralleldots and Repustate as case studies. We adopted benchmark Arabic social networks data collections, simulate the user interaction with the ParallelDots AI API and Repustate API and classified the corpus into positive, negative or neutral polarity classes. In our experiments we evaluated the results of these online tools and seven machine learning classifiers for all polarity classes; positive, negative and neutral. We highlighted the limitations and weaknesses for every polarity class and we compared several classifiers for Arabic sentiment analysis. The results showed that there are limitations in detecting neutral class for all the tested solutions. Depending on our obtained outcomes we are highly recommending use Paralleldots API for Arabic sentiment analysis for both positive and negative reviews. As a future work we plan to assess multimodal Arabic sentiment analysis, extend the data collection and evaluate other online tools.

ACKNOWLEDGEMENT

This research is funded by the Deanship of Research and Graduate Studies in Zarqa Private University / Jordan.

REFERENCES

[1] S. Poria, A. Hussain and E. Cambria, **Multimodal Sentiment Analysis**, *Socio-Affective Computing*, Springer, 2018. https://doi.org/10.1007/978-3-319-95020-4

[2] M. Al-Kabi, H. Wahsheh and I. Alsmadi, **Polarity Classification of Arabic Sentiments**, *International Journal of Information Technology and Web Engineering*, Vol. 11, No. 3, pp. 32-49, 2016.

[3] F. Salem, **Social Media and the Internet of Things towards Data-Driven Policymaking in the Arab World: Potential, Limits and Concerns**, *The Arab Social Media Report, Dubai: MBR School of Government*, Vol. 7, 2017.

[4] H. Khafajeh, N. Yousef and M. Abdeldeen, **Arabic root extraction using a hybrid technique**, *International Journal of Advanced Computer Research*, Vol. 8, No. 35, pp. 89-95, 2018.

[5] M. Al-Kabi, N. Al-Qudah, I. Alsmadi, M. Dabour and H. Wahsheh, **Arabic/English sentiment analysis: an empirical study**, *Proc. Fourth International Conference on Information and Communication Systems (ICICS 2013)*, pp.1-6, 2013.

[6] S. Salloum, A. AlHamad, M. Al-Emran and K. Shaalan, **A Survey of Arabic Text Mining**, *Intelligent Natural Language Processing: Trends and Applications*, pp. 417-431, 2017.

[7] M. Alrefai, H. Faris and I. Aljarah, **Sentiment Analysis for Arabic Language: A Brief Survey of Approaches and Techniques**, *International Journal of Advanced Science and*

Technology, Vol. 119, pp. 13-24, 2018.
https://doi.org/10.14257/ijast.2018.119.02

[8] F. Mahyoub, M. Siddiqui and M. Dahab, **Building an Arabic Sentiment Lexicon Using Semi-supervised Learning**, *Journal of King Saud University - Computer and Information Sciences*, Vol. 26, No. 4, pp. 417-424, 2014.

[9] **Sentiment Analysis | ParallelDots AI APIs**, Paralleldots.com, [Online]. Available: https://www.paralleldots.com/sentiment-analysis, [Accessed: 03- Nov- 2019].

[10] **Repustate Sentiment Analysis API**, [Online]. Available: https://www.repustate.com/sentiment-analysis-api-demo/, [Accessed: 03- Nov- 2019].

[11] M. Al-Kabi, M. Al-Ayyoub, I. Alsmad, and H. Wahsheh, **A prototype for a standard Arabic sentiment analysis corpus**, *The International Arab Journal of Information Technology (IAJIT)*, Vol. 13, pp. 163-170, 2015.

[12] M. Abdul-Mageed, S. Kübler and M. Diab, **SAMAR: a system for subjectivity and sentiment analysis of Arabic social media**, *Proc. 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis (WASSA '12)*, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 19-28, 2012.

[13] E. Refaee and V. Rieser, **Can we Read Emotions from a smiley face? Emoticon-based distant supervision for subjectivity and sentiment analysis of Arabic Twitter feeds**, *Proc. 5th International Workshop on Emotion, Social Signals, Sentiment and Linked Open Data*, Reykjavik, Iceland, pp. 1-5, 2014.

[14] A. Mourad and K. Darwish, **Subjectivity and Sentiment Analysis of Modern Standard Arabic and Arabic Microblogs**, in *Proc. 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Atlanta, Georgia, pp. 55–64, 2013.

[15] B. Al Shboul, M. Al-Ayyoub and Y. Jararweh, **Multi-way sentiment classification of Arabic reviews**, in *Proc. 6th International Conference on Information and Communication Systems (ICICS)*, pp. 206-211, 2015.
https://doi.org/10.1109/IACS.2015.7103228

[16] A. Shoukry and A. Rafea, **Sentence-level Arabic sentiment analysis**, in *Proc. IEEE International Conference on Collaboration Technologies and Systems (CTS)*, 2012 pp. 546-550, 2012.

[17] A. M. Rabab'ah, M. Al-Ayyoub, Y. Jararweh and M. Al-Kabi, **Evaluating sentistrength for arabic sentiment analysis**, in *Proc. 7th IEEE International Conference on Computer Science and Information Technology (CSIT)*, pp. 1-6, 2016.

[18] R. Bose, R. Dey, S. Roy and D. Sarddar, **Analyzing Political Sentiment Using Twitter Data** *In Information and Communication Technology for Intelligent Systems*, pp. 427-436, Springer, Singapore, 2019.

[19] N. Abdulla, N. Ahmed, M. Shehab and M. Al-Ayyoub, **Arabic sentiment analysis: Lexicon-based and corpus-based**, in *Proc. 2013 IEEE Jordan conference on applied electrical engineering and computing technologies (AEECT)*, pp. 1-6, 2013.

[20] M. Skirpan and C. Fiesler, **Ad empathy: A design fiction,** *Proc. 2018 ACM Conference on Supporting Groupwork*, pp. 267-273. ACM, 2018.

[21] M. Gharibi, **Building a Knowledge Graph for Food, Energy, and Water Systems**, University of Missouri--Kansas City, pp. 1-53, 2017.

[22] Ç. Erion, **Cloud-based Recommendation Systems: Applications and Solutions**, pp. 1-5, 2019.

[23] J. Li, S. Ji, T. Du, B. Li and T. Wang, **TextBugger: Generating Adversarial Text Against Real-world Applications**, in *Proc. Network and Distributed Systems Security (NDSS) Symposium*, pp. 1-15, 2018.

[24] P. Malhotra, L. Vig, G. Shroff and P. Agarwal, **Long short term memory networks for anomaly detection in time series**, in *Proc. ESANN, 23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 89-94, 2015.

[25] H. Wahsheh, M. Al-Kabi and I. Alsmadi, **SPAR: A system to detect spam in Arabic opinions**, in *Proc. 2013 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, pp. 1-6, 2013.

[26] H. Wahsheh, I. Alsmadi and M. Al-Kabi, **Analyzing the popular words to evaluate spam in Arabic web pages**, *IJJ: The Research Bulletin of JORDAN ACM–ISWSA*, Vol. 2, No. 2, pp. 22-26, 2012.

[27] H. A. Wahsheh, M. N. Al-Kabi and I. M. Alsmadi, **Spam detection methods for Arabic web pages**, in *Proc. First Taibah University International Conference on Computing and Information Technology-Information Systems (ICCIT)*, pp. 486-490, 2012.

[28] H. A. Wahsheh, M. N. Al-Kabi and I. M. Alsmadi, **A link and content hybrid approach for Arabic web spam detection**, *International Journal of Intelligent Systems and Applications (IJISA)*, Vol. 5, No.1, pp. 30-43 , 2013.
https://doi.org/10.5815/ijisa.2013.01.03

[29] I. Witten and E. Frank, ***Data Mining: Practical Machine Learning Tools and Techniques***, Morgan Kaufmann Series in Data Management Systems, pp. 1-558, 2005.

[30] A. Moore, **Support vector machines**, Tutorial. School of Computer Science of the Carnegie Mellon University. http://www. cs. cmu. edu/˜ awm/tutorials, [Accessed 16-Aug-2019].

[31] M. L. Zhang and Z. H. Zhou, **A k-nearest neighbor based algorithm for multi label classification**, in *Proc. 2005 IEEE International Conference on Granular Computing*, Vol. 5, pp.718-721, 2005.

[32] P. Strecht, L. Cruz, C. Soares, J. Mendes-Moreira and R. Abreu **A Comparative Study of Classification and Regression Algorithms for Modelling Students' Academic Performance**, in *Proc. 8th International Conference on Educational Data Mining*, pp. 392-395, 2015.

[33] **Web.csulb.edu**, 2019. [Online]. Available: https://web.csulb.edu/~tebert/teaching/lectures/551/random_forest.pdf [Accessed: 06- Mar- 2019].

[34] M. Govindarajan, **Ensemble of Classifiers in Text Categorization**, *International Journal of Emerging Trends in Engineering Research*, Vol.8 No. 1, pp. 41-45, 2020.
https://doi.org/10.30534/ijeter/2020/09812020

[35] B. G. Becker, **Visualizing decision table classifiers**, in *Proc. IEEE Symposium on Information Visualization* (Cat. No. 98TB100258), pp. 102-105, 1998.

[36] T. R. Patil and S. S. Sherekar, **Performance analysis of Naive Bayes and J48 classification algorithm for data classification**, *International journal of computer science and*

applications, Vol. 6, pp. 256-261, 2013.
https://doi.org/10.30534/ijeter/2019/177112019

[37] P. Hooda, and P. Mittal. **An exposition of data mining techniques for customer churn in telecom sector**. *International Journal of Emerging Trends in Engineering Research*, Vol. 7, No. 11, pp. 506-511, 2019.

[38] A. Dhankhar, and K. Solanki. **A comprehensive review of tools and techniques for big data analysis**. *International Journal of Emerging Trends in Engineering Research*, Vol. 7, No. 11, pp. 556-562, 2019.
https://doi.org/10.30534/ijeter/2019/257112019