Towards Analysing the Sentiments in the Field of Automobile with Specific Focus on Arabic Language Text

Ayman Yafoz and Malek Mouhoub

Department of Computer Science, University of Regina, 3737 Wascana Parkway, Regina, Canada

Keywords: Natural Language Processing, Sentiment Analysis, Arabic Language, Automobile.

Abstract: Performing natural language processing on a text document is an active research area. Social media includes valuable information resources in various languages. These information resources include reviews, comments, tweets, posts, opinions, articles and other text resources which could be analysed to explore people's opinions, attitudes, emotions and sentiments towards various subjects and commodities. However, there is lack of contributions addressing sentiment analysis on automobile reviews in Gulf Cooperation Council (GCC) dialects and in the Arabic language in general. Moreover, the lack of available annotated datasets in the Arabic language, which are targeting specific domains (such as automobiles), and the limited focus on analysing the sentiments in Arabic regional dialects created a gap. These factors motivated us to conduct the research that we report in this paper. Furthermore, the limited adoption of techniques and algorithms related to natural language processing and machine learning is noticed in the current efforts that are targeting sentiment analysis in the Arabic language, which once adopted could provide the opportunity of enhancing the performance of sentiment analysers. Therefore, this research attempts to cover these gaps to a certain extent.

1 INTRODUCTION

The purpose of sentiment analysis (aka opinion mining) is to analyse the opinions of people towards evaluations, attitudes, sentiments, opinions, emotions and appraisals towards objects such as individuals, topics, issues, services, organizations, services, products, events and their features, which constitute a large research area. Several terms can be used to refer to sentiment analysis such as opinion extraction, subjectivity analysis, emotion analysis, opinion mining, sentiment mining, review mining and affect the analysis. However, all the previous terms are either under the sentiment analysis umbrella or the term opinion mining (Liu, 2012).

Moreover, the area of sentiment analysis is a key research area in the field of applied linguistics. The importance of performing sentiment analysis is noticed in fields such as politics, education and marketing. Furthermore, opinion mining or sentiment analysis extracts information through determining the data that indicate neutral, negative or positive texts in given documents (Zaidan et al,2014). This extraction could be accomplished through the fields of machine learning, NLP and statistics that assist in determining the polarity of a given record (Zaidan et al, 2014). The extracted critique, feedback or comment might contain sentiments that could provide a valuable indicator for several purposes.

There are three sentiment classes to which the extracted sentiment is classified: negative, positive or neutral; or it could be classified using an n-point scale such as very bad, bad, satisfactory, good or very good. Each of the previous classes forms a sentiment. This process provides means to the corporations to evaluate the acceptance rate of their products in the market and assist them in creating plans to enhance the quality of their products.

Sentiment analysis could also assist politicians or policymakers in evaluating the sentiments related to political matters, public services or policies (Prabowo et al, 2014). This process is also currently important due to the proliferation of social media where the analysis of the feelings and emotions of social media users is crucial for many marketers, other parties in the social media industry and external agencies.

Due to the importance of analysing and understanding reviewers' feedback and comments to fulfil their needs and understand the situation in the market, this contribution comes to address these issues through analysing the sentiments of online reviews in the Arabic language, which are exclusively related to automobiles. The novelty of this research lies in filling the gap of lacking a specific sentiment

Yafoz, A. and Mouhoub, M.
Towards Analysing the Sentiments in the Field of Automobile with Specific Focus on Arabic Language Text.
DOI: 10.5220/0008916904030412
In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence (ICAART 2020) - Volume 2*, pages 403-412
ISBN: 978-989-758-395-7; ISSN: 2184-433X

analyser that works on a dataset related to automobiles and in both modern standard Arabic and GCC dialect. Moreover, the expressions that convey sentiments in the automobile domain are rarely found in other domains, for instance, the expression “high fuel consumption” reflects a negative sentiment, while the expression “ice-cold AC” reflects a positive sentiment. Therefore, the research aims to perform sentiment analysis processes on a dataset that is exclusively related to the automobile industry.

The novelty extends to cover the methodology proposed by this research to address that problem being considered, which is reflected in the accuracy outcomes of the proposed system (nearly 84% as shown in the experiment evaluation results we report in this paper). The system is composed of seven main stages, which are described briefly below:

1- Data Gathering: in this stage, the data related to automobiles is gathered from the websites “Haraj” and then filtered to leave data containing sentiments (positive, negative or mixed) in the dataset. For this task, a dedicated web scrapper is developed using the Beautiful Soup, which is a Python library.
2- Data Annotation: in this stage, the data is annotated by three annotators and following that the inter-rater agreement is computed to show the consistency and agreement level between annotators.
3- Data Pre-processing: in this stage, the data undergo a cleansing process to ensure it does not contain irrelevant data and is suitable for classification.
4- Feature Selection: the purpose of feature selection is to find the most informative and compact group of features of a particular task to enhance processing, data storage or efficiency which is important for classification, regression and clustering in both unsupervised or supervised approaches. In feature selection, the initial features are decreased and a subgroup retaining sufficient information for acquiring satisfactory or better performance outcomes is chosen (Bolón-Canedo et al, 2015).
5- Splitting the Dataset: in this phase, the dataset is divided into training and testing datasets. The unbalanced dataset is treated to avoid inadequate representation of minority classes in the training dataset.
6- Data Processing: in this stage, a group of machine learning classifiers will perform the classification process.
7- Data Visualization: in this stage, the word cloud will visually depict the most frequent words and sentiment words in the dataset.

Moreover, the dataset is sourced from the famous GCC forum website called “Haraj”, which contains reviews related to different aspects of automobiles. Haraj is one of the most common websites in Saudi Arabia that encompasses topics related to automobiles in Saudi Arabia specifically. A GCC dialect was mostly used by the reviewers to write the topics. The dataset gathered from “Haraj” contains around 6,585 comments divided into three categories (positive, negative and mixed opinions). The dataset focuses on almost 29 topics related to automobiles, which are illustrated in Table 1:

Table 1: Covered Topics in Automobile Dataset.

Air-Conditioning	Spare Parts	Quality	Gasoline Consumption
Agents	Design and Shape	Sales and Proliferation	Comfort
Accessories and Specs	Performance and Practicality	Maintenance	Price and Resale Value
Sounds	Lubricants	Powertrain	Safety
Breadth	Malfunctions	Luxury	Power
Gearbox	Insulation	Temperature	Vision
Insurance	Roughness	Stability and Torque	

The following example in Table 2 illustrates the predicted results of a sentiment analysis process on comments extracted from the dataset:

Table 2: The Predicted Results of a Sentiment Analysis Process.

Sentence	Translation	Sentiment
الموستنج سيارة جميلة و تنفع للإستعمال اليومي وقطعها أرخص من منافسيها.	Mustang is a beautiful car that works for everyday use, and its parts are cheaper than its competitors.	Positive
الاكورد مستبعدة بسبب سعرها المبالغ فيه على 4 سلندر و قير CVT.	The Accord is excluded due to its exaggerated price on 4 cylinders and due to CVT.	Negative
التورس سيارة مرغوبة جدا لكن مشكلتها إنها ضيقة إلى حد ما من الداخل.	The Taurus is a very desirable car, but its problem is that it is somewhat tight inside.	Mixed

In this contribution, Python is selected to write the program codes due to its popularity, libraries (such as NLTK, Sklearn, Pandas and so forth), easiness and the framework it provides to write codes for sentiment analysis tasks. On the other hand, to save the dataset, MySQL Community Server is selected because it is compatible with Windows operating system and can be connected with Python projects.

The rest of the paper describes each of the system stages we listed before. It then reports on the experimental evaluation that we have conducted to assess the performance, in terms of accuracy, of the proposed system. Finally, concluding remarks are then presented in the last section.

2 DATA COLLECTION, ANNOTATION AND PRE-PROCESSING

2.1 Data Gathering and Annotation

Annotation is the process of adding labels to pictures, audios, videos, text, records or documents as notes, external remarks, comments or explanations without modifying the labelled item (Dingli, 2011). In this contribution, the annotation referred to the sentiment labels of the record in the dataset (positive, negative or mixed).

Moreover, after creating the annotation guidelines and distributing them with the dataset to three annotators who performed the annotation and returned a completely labelled dataset. We gathered the dataset from each annotator and computed the interrater agreement scores using Fleiss Kappa formula:

$$k = \frac{\overline{P} - \overline{P}_e}{1 - \overline{P}_e} \quad (1)$$

The formula part $\overline{P} - \overline{P}_e$ denotes the actually obtained agreement degree above chance, where $1 - \overline{P}_e$ denotes the agreement degree that is achievable above chance. The case where $k \leq 0$ reflects the lack of agreement among assessors, and the situation where $k = 1$ reflects a total agreement among assessors (Nichols et al, 2010). Moreover, in this project, the Fleiss Kappa degree = 82.7% which reflects a substantial agreement among interrater.

2.2 Cleaning the Data

Cleaning the data referred to the operation of adjusting anomalies in the dataset, such as correcting typographical mistakes, duplicated or missing words or letters and formatting variances, which is important for data mining and natural language processing tasks (Manolescu et al, 2007) as it facilitates and accelerates processing the data by machine learning algorithms (Squire, 2015).

In this research, the eastern Arabic numerals were converted to Arabic numerals to enhance the text consistency, and the stretched words were shortened and vice versa to achieve the most optimal words' form. Furthermore, grammar mistakes were corrected, and some punctuations were either deleted, added or replaced with the correct punctuations.

Moreover, some clauses were rearranged to obtain the correct words' order. Additionally, cases of missing, redundant or incorrect mixing of female and male pronouns were resolved as well as redundant and missing words were addressed. Furthermore, the issue of the wrongly used form of singular or plural words was handled to achieve the correct form of a word. Furthermore, the numbers in the numbered list were deleted and the remaining text was merged.

Moreover, incorrectly separated words were merged, and mistakenly joint words were detached. Furthermore, the spaces between the words and the letters were either added or removed based on each situation to achieve the correct form. Moreover, extremely racist, personal insults, offensive and aggressive reviews were excluded from the dataset.

Furthermore, the cases of superfluous and lost prepositions, conjunctions and the Arabic article (ال) were handled, and exclamation and question marks were removed from the text. Moreover, the records containing similar reviews were deleted to ensure the uniqueness.

In addition to that, numeric ratings (such as 10/10) were converted to alphabetic rating. Furthermore, diacritics were deleted from the text, and some punctuations were converted to prepositions. Moreover, the plus sign "+" was converted to conjunction. In addition to that, emojis, hashtags, links, ellipsis, hyphens, guillemets and asterisks. Semicolons and slashes were removed to attain a clean dataset.

2.3 Regular Expression

As part of the normalization process, an automatic regular expression process is adopted to ensure further cleansing of the data and make the data files only contain Arabic letters in the correct form. The benefits of using regular expression technique in NLP projects is to validate the text in the records, filter the data, and find and replace operations on the data (Kaur, 2014).

In this research, the regular expression process starts with finding the English alphabets in the dataset records and replacing them with spaces. Moreover, the regular expression process replaces all the forms of the Arabic letters “أ” and “ي” with the letters “ا” and “ى” to reduce ambiguate situations that could arise in the classification processes. Furthermore, all the numbers in the text are excluded as they do not convey sentiments, and all the diacritics are automatically removed again as they arguably not affecting the sentiment category of the text.

Moreover, another preparing process is adopted to enhance the filtration process on the data, which is to stop word removal. A list containing around 760 stop words was prepared for this process. These stop words will be excluded from the data records in both the dataset and the dictionary as these words are considered redundant words lacking valuable semantics that should be eliminated to reduce the data volume, accelerate the processing and enhance the accuracy of the classification (Raulji, 2016). Therefore, each word in both the dataset and the dictionary is tokenized to filter out every stop word. Following that, the tokenized words are joint together again as filtered whole sentences.

2.4 Stemming

A stemming approach is applied on the data. Stemming of the words is a crucial part in both information retrieval and NLP systems. The current search systems and indexing support stemming feature. The purpose of stemming is to enhance recalling by treating word endings by returning the words to their roots. The recall process should be enhanced by performing stemming while considering that the accuracy of fetching the records is not deteriorated (Jivani, 2011).

Moreover, typical stemming includes the exclusion of prefixes and suffixes that are attached to the index words prior to assigning the word to the target index. This stemmed word reflects a wider notion in comparison with the original word, and the number of retrieved records by the information retrieval system will increase as a result of performing the stemming operation. Moreover, this stemming process is important for summarization, categorization and text clustering as it is a component of pre-processing operation required before performing any classification algorithm (Jivani, 2011). Table 3 shows an example of using the stemmers Light 10 and P-Stemmer (Kanan, 2016).

Therefore, this contribution adopted The Information Science Research Institute’s (ISRI) Arabic stemmer. This stemmer h as several mutual features with Khoja stemmer (Kanan, 2016). Nevertheless, ISRI stemmer is lacking a root dictionary as its developers claim that it does not enhance the retrieval of documents only containing a monolingual Arabic language. The ISRI stemmer performed better than both light and Khoja stemmers on the queries having shorter titles (Elkhoury, 2005).

Table 3: Using the Stemmers Light 10 andP-Stemmer on Arabic Text (Kanan, 2016).

Word	Light 10	P-Semmer
كالصادرات As the Imports	صادر Took	صادرات Imports
والوحدات And the Units	وحد Aggregate	وحدات Units
المكتبات The Libraries	مكتب Office	مكتبات The Library
المباحثات The Negotiaitons	مباحث Investigation	مباحثات Negotiations

3 FEATURE SELECTION

Many researchers argue that adopting a feature selection approach in natural language processing projects would enhance the accuracy and efficiency of the classification process (Mohod, 2014). Feature selection reduces the dimensionality of the dataset to reduce processing costs, reduce memory load and extract more beneficial information through summarizing the dataset by finding finer matrices of narrower attributes and samples in the original dataset matrix, which could be more beneficial for the classification purpose than the original dataset (Bolón-Canedo, 2015).

3.1 The Part-Of-Speech Tagger (POS Tagger)

The part of speech (POS) tagging process is defined as the computational identification of which part of speech is triggered by using a word in a specific text. The POS tagging process plays a vital role in the pre-processing phase in numerous NLP projects, such as named entity recognition, speech processing, and sentiment analysis as it affects the results of the consequent data processing phases. The POS tagger allocates a specific part of speech to each word in a given sentence, which could enhance the performance (Jivani, 2011) of certain sentiment analysis approaches. Moreover, Table 4 shows an example of performing part of speech tagging on Arabic text.

Table 4: An example of Applying Part of Speech Tagger on Arabic Text.

Word	Translation	POS
سيارة	Car	Noun
سريعة	Fast	Adjective
للأسف	Sadly	Adverb
تعمل	Works	Verb

Therefore, a POS tagger is applied in this contribution to assist the classifier in distinguishing each word type and focus on analysing the word types with shared POS tags that hold sentiments and are most important in the classification process. This reduces the load of processing and enhances the accuracy of processing data in sentiment analysis tasks not only in the Arabic language but in other languages (Gusev, 2018).

3.2 Term Frequency-Inverse Document Frequency

One of the most used feature selection approaches in sentiment analysis is TFIDF (Term Frequency-Inverse Document Frequency). TFIDF is a statistical approach that indexes the terms. In term frequency of the TFIDF, all the records are defined as vectors that include words such as (Deshmukh, 2016):

R = <Word1, Word2, Word3,……., Wordn> (2)

Where R refers to the record and word is the term in that record and n reflects the count of words in that record (Deshmukh, 2016).

TFIDF depends on term vectors and documents representing both term presence and term frequency. The term frequency is used to calculate the terms in TFIDF. The most common terms in a specific document have higher term frequency values. Any term that excessively appears in numerous documents would be ignored (Deshmukh, 2016). Therefore, the purpose of calculating TFIDF is to determine each word's importance in the sentence through normalizing the term frequency in a record as for the whole dataset (Ojeda, 2018). This gives an advantage of TFIDF over other statistical approaches, such as bag of words (BOW) that treats all terms in the document equally regardless (Ojeda, 2018) of their importance to the sentiment classification process (such as articles and conjunctions).

The value of TFIDF as stated above is the result of multiplying the term frequency (TF) related to a particular term in a document with the inverse document frequency related to this term in the whole dataset. The mathematical representation of TFIDF is (Ghosh, 2018):

$$\mathrm{TF.IDF} = tf_d^{\,i} * log\,\frac{N}{df^i} \qquad (3)$$

In the above equation, $tf_d^{\,i}$ refers to the count of times that a given term of i appeared in document $d.$ N represents the total count of documents in a dataset, and df^i is the count of documents that the term i appears in. Moreover, TF represents the term's importance in a given document; the higher the value of TF the more frequently the term occurs in the document. IDF denotes the extent to which a specified term is informative in the dataset (Ghosh, 2018).

In this contribution, a TFIDF vectorizer is applied to the dataset with three thresholds. The first threshold "min_df" excludes too infrequent terms. It excludes all the terms that occur in less than ten records. The second threshold "max_df" excludes too frequent terms. It excludes all the terms that occur in more than 75% of the records (which in this case could be articles, stop words or terms unimportant to sentiment classification). The third threshold is "ngram_range", which sets the upper (which is 3) and lower (which is 1) boundaries of the n-values range for various n-grams that will be extracted (TfidfVectorizer).

3.3 N Gram Feature

N gram reflects a group of words appearing in a dataset. The main purpose of N gram is to create features that are going to be used by supervised machine learning algorithms (Kshirsagar et al, 2016) (Bhayani et al, 2009). Moreover, research conducted by (Bhayani et al, 2009) state that using bigrams assisted the classifier to a large extent in detecting negated expressions as explicit features (such as "not powerful") more than unigrams.

However, the same research claims that using bigrams only is not efficient and should be combined with unigrams as features, as the feature spaces of bigrams is highly sparse (sparsity is defined as not finding sufficient data in a dataset to comprise a language correctly (Allison et al, 2006)). Therefore, the higher the number of n-grams (for instance, five or four grams) could arguably generate more sparsity and incorrect results (Allison et al, 2006). However, including bigrams and trigrams to select features would arguably enhance the process of capturing sentiments in corpus (Narayanan et al, 2013).

Based on the above, the n-grams of unigrams, bigrams and trigrams were adopted in this contribution to enlarge the scope of feature selection with the purpose to capture more sentiments in the text.

4 SPLITTING THE DATASET

The large volume of the dataset induced the adoption of machine learning models to both extract the data and predict the model. For these two aims, it became widespread to partition the dataset into training and testing combinations of data. Moreover, the training set is created to learn the machine learning model, while the testing set is employed later to assess the function of the model trained on the training set. Many present projects in machine learning divide the dataset into roughly 70% for training set and the rest 30% for testing set (Cocea et al, 2017), and therefore, the dataset in this contribution is partitioned into 70% for training set and 30% for the testing set to ensure attaining an adequate training and testing performance.

4.1 Cross Validation

This validation approach randomly splits the training dataset into k subsections and 1 of the k–1 subsections will be allocated for testing purposes while the rest will be trained on. The favourite value for k is 10- fold cross-validation that is adopted by many validations in machine learning. This reflects that the nine subsections will be allocated for training the classifier and the rest 1 is specified for testing processes. In this contribution, cross-validation is adopted to prevent overfitting from occurring in the training set, which is common in small datasets and abundant attributes (Bazazeh et al, 2016).

4.2 Synthetic Minority Over-sampling Technique Nominal Continues

The synthetic minority over-sampling technique (SMOTE) is one of the important and powerful techniques to perform oversampling on the minority classes (Xu et al, 2006). SMOTE could arguably enhance the performance of the classifier without losing valuable data as it produces extra samples of minority classes which enable the classifier to enhance the learning process and broaden the coverage and decision areas. Therefore, SMOTE is widely used by the researchers in the software field due to its outstanding outcomes (Pears et al, 2014) and its capability of providing better performance than the ordinary oversampling approach (Blagus et al, 2012).

Moreover, because SMOTE technique can not work with datasets that are mainly composed of nominal “categorical” features (Kalajdziski et al, 2015), a generalization of SMOTE technique is developed to work with datasets containing a mix of nominal and continuous numeric (Precision_Recall_Fscore_Support) features, this generalized technique is named SMOTE-NC “Synthetic Minority Over Sampling Technique Nominal and Continuous”. In this contribution, SMOTE-NC is adopted to perform the synthetic oversampling process for the minority classes in the training dataset.

5 EXPERIMENTATIONS

In this experiment, twenty-two machine learning algorithms were adopted to perform the classification task. The results of the classification processes are illustrated in Table 5 (see Table 5 for the list of the classifiers used in these experiments). A classification report is generated after every classification process. The classification report presents the computations of precession, recall, f1-score and support. The precision is the ratio of:

$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} = \frac{\text{True Positive}}{\text{Total Predicted Positive}} \tag{4}$$

The precision refers to measuring the extent to which it is affirmed that the classifier will not label a positive sample as a negative one (Precision_Recall_Fscore_Support). The optimal precision value is 1 and the poorest is 0.

The recall is:

$$Recall = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} = \frac{\text{True Positive}}{\text{Total Actual Positive}} \tag{5}$$

The recall refers to measuring the extent to which the classifier is capable of detecting all incidents of positive (negative or mixed) samples (Precision_Recall_Fscore_Support). The optimal recall value is 1 and the poorest is 0. Furthermore, the F1-score, which is also called F measure or F-score has the ratio of:

$$F1\ Score = 2 * \frac{\text{Precision} \ * \ \text{Recall}}{\text{Precision} \ + \ \text{Recall}} \tag{6}$$

The F1-score in the previous equation can be regarded as the weighted average of both precision and recall. The optimal F1-score value is 1 and the poorest is 0 (Precision_Recall_Fscore_Support). Moreover, the score of the criterion support is resulted from counting how many times each class "label" in y_true "the training" has occurred (the number of positive, negative and mixed incidents supported the classifier that occurred in the training dataset) (Precision_Recall_Fscore_Support).

Furthermore, the field of weighted average computes metrics for every label along with finding the average weighted based on support for every label "the count of true cases for every label." The average weighted adjust macro to consider imbalance label and could produce F-score, which is not between recall and precision (Precision_Recall_Fscore_Support). Hence, the support-weighted mean is averaged for each label (The Classification Report). Furthermore, the following is the mathematical equation (Barendregt et al, 1986) (Weighted Averaging) of the weighted average:

$$Weighted\ Average = \frac{\sum_p H_p T_p}{\sum_p H_p} \tag{7}$$

Where H_p reflects the support for a sentiment polarity p (positive, negative and mixed), and T_p denotes the precision for a sentiment polarity p. The result of summing this value is divided by the overall sum of all sentiment polarities support $\sum_p H_p$.

The accuracy of a machine learning algorithm refers to the probability ratio of predicting the category of unlabeled records. Hence, the generalization ability of a classifier can be reflected through the index of its high accuracy rates on unseen records that were tested in the testing dataset (Tagliaferri et al, 2018). Moreover, the following equation illustrates the computation of the accuracy metric (Grus, 2015):

$$Accuracy = \frac{\text{Correct (True Positive+True Negative)}}{\text{Total (True Positive+True Negative+False Positive+False Negative)}} \tag{8}$$

Furthermore, in this contribution, the accuracy is computed for the ten folds created by the cross-validation approach, then the average of these accuracy calculations is computed as shown in the following equation:

$$\text{Accuracy of the Classifier} = \frac{\text{Toatl of all 10 Accuracy Resluts for the cross−validation folds}}{\text{Number of Cross−Validation Folds (10)}} \tag{9}$$

Based on the results in Table 5, in terms of the average of accuracy results, the Ensemble Hard Vote classifier outperformed all other adopted classifiers by scoring approximately 84%. Moreover, in terms of the precision and recall weighted averages, the Ensemble Hard Vote, the Ridge, and the Ensemble Soft Vote classifiers outperformed all other adopted classifiers by scoring 84% each. Furthermore, in terms of the F1-score weighted averages, the Ensemble Hard Vote, the Ridge, the Ensemble Soft Vote, the Linear Support Vector and the Logistic Regression CV classifiers outperformed all other adopted classifiers by scoring 83% each. Hence, the results reflect that the Ensemble Hard Vote classifier should be adopted to analyse the sentiments in Arabic automobile datasets due to its performance in four measured scales.

Moreover, a calculating statistical evaluator called a confusion matrix, which is composed of false positive, true positive, false Negative and True Negative is adopted to evaluate the correctness of the conducted classifications. More precisely, in this contribution, the confusion matrix is calculated with and without normalization. The purpose of normalizing the confusion matrix is to make the sum of the values in every sentimental field of positive, negative and neutral equal to one (Nkambou et al, 2012). Therefore, the values in each of the three sentimental categories fields (positive, negative and neutral) will be ranging between 0 and 1 as shown in Figure 1.

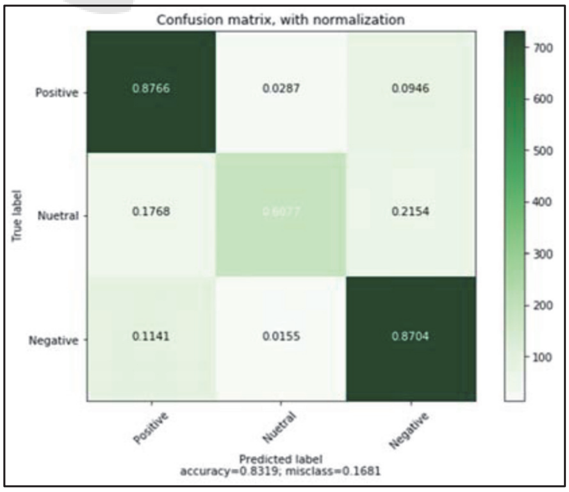

Figure 1: Confusion Matrix, with Normalization.

Table 5: A Comparison Between the Machine Learning Algorithms in terms of the Average of Accuracy, Precision Weighted and Recall Weighted Results.

NO.	Name of the Machine Learning Algorithm	Average of Accuracy Results	Precision Weighted Average	Recall Weighted Average	F1-Score Weighted Average
1	The Ensemble Hard Vote Classifier	83.79%	84%	84%	83%
2	The Ridge Classifier	83.54%	84%	84%	83%
3	The Ensemble Soft Vote Classifier	83.49%	84%	84%	83%
4	Linear Support Vector Classifier	83.19%	83%	83%	83%
5	Logistic Regression CV Classifier	82.89%	83%	83%	83%
6	The Ridge CV Classifier	82.79%	83%	83%	82%
7	Stochastic Gradient Descent Classifier	81.83%	83%	83%	82%
8	The Bernoulli Naive Bayes Classifier	81.48%	81%	81%	81%
9	Logistic Regression Classifier	81.23%	82%	81%	80%
10	Passive Aggressive Classifier	80.72%	81%	81%	81%
11	The Nearest Centroid Classifier	80.07%	80%	80%	80%
12	The Perceptron Classifier	79.01%	79%	79%	79%
13	Extra Trees Classifier	78.82%	79%	79%	78%
14	The Multinomial Naive Bayes Classifier	78.06%	79%	78%	75%
15	The Multi Layer Perceptron Classifier	78.06%	78%	78%	78%
16	Gradient Boosting Classifier	77.60%	78%	77%	77%
17	Bagging Classifier	76.55%	78%	78%	77%
18	Random Forest Classifier	76.54%	76%	75%	75%
19	Decision Tree Classifier	74.23%	74%	74%	74%
20	Ada Boost Classifier	68.95%	72%	69%	68%
21	K-Nearest Neighbors Classifier	63.35%	67%	63%	62%
22	Support Vector Classifier	42.33%	18%	42%	25%

5.1 Data Visualization

To graphically depict the most frequent words and sentiment terms in the text, a technique named word cloud is adopted in this contribution. The expression word cloud reflects creating a file summary through mining the most recurrent terms in a given file (Franconeri et al, 2018).

Moreover, the word cloud could be defined as a weighted list created to depict text data, and recently it attracted many researchers in big data field that adopt them to graphically represent the data in an attractive and easy to understand manner.

However, the use of the word cloud to represent non-English words is arguably limited (Jin, 2017), which justifies the adoption of the word cloud in this contribution. Consequently, Figure 2 shows a word cloud presenting the most frequent sentiment words in the dataset.

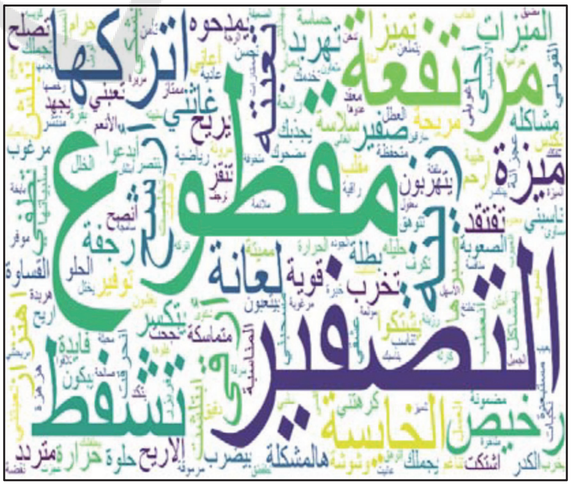

Figure 2: A Word Cloud Illustrating the Most Frequent Sentiment Words in the Dataset.

6 CONCLUSION

This paper presents a framework for analysing the sentiments in the Arabic language that is related to the automobile field. The paper also explained the details of the project phases, data collection, annotation procedures, how the data is cleaned, the feature selection process, the way of splitting the data, how the data is graphically depicted and the classification process with the results of the twenty-two machine learning classifiers adopted in this contribution. The highest obtained result for accuracy is 83.79% by the Ensemble Hard Vote classifier. Hence, the results reflect that the Ensemble Hard Vote classifier should be adopted to analyse the sentiment in Arabic automobile datasets due to its high results in the four measured scales.

In future work, more experiments and studies will be conducted on how to enhance the accuracy results through improving the cleaning process, including dictionary as a hybrid approach and adopting advanced deep learning algorithms.

REFERENCES

Liu, B., 2012. *Sentiment Analysis and Opinion Mining*, Morgan & Claypool Publishers.

Zaidan, O., Callison-Burch, C., 2014. Arabic Dialect Identification. *Computational Linguistics Journal*. 40(1).

Prabowo, R., Thelwall, M., 2009. Sentiment Analysis: A Combined Approach. *Journal of Informetrics*. 3(2).

Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A., 2015. *Feature Selection for High-Dimensional Data (Artificial Intelligence: Foundations, Theory, and Algorithms)*. Springer. 1st Edition, 2015 Edition.

Dingli, A., 2011. *Knowledge Annotation: Making Implicit Knowledge Explicit*. Springer. 2011 Edition.

Nichols, T., Wisner, P., Gulabchand, L., Cripe, G., 2010. Putting the Kappa Statistic to Use. *The Quality Assurance Journal*. 13(3-4).

Manolescu, I., Weis, M., 2007. Declarative XML Data Cleaning with XClean. In *CAiSE 2007, International Conference on Advanced Information Systems Engineering*.

Squire, M., 2015. *Clean Data*. Packt Publishing.

Kaur, G., 2014. Usage of Regular Expressions in NLP. In *IJRET*, *International Journal of Research in Engineering and Technology*. 3(4).

Raulji, J., Saini, J., 2016. Stop-Word Removal Algorithm and its Implementation for Sanskrit Language. *International Journal of Computer Applications*. 150(2).

Jivani, A., 2011. A Comparative Study of Stemming Algorithms. *International Journal of Computer Technology and Applications*. 2(6).

Kanan, T., Fox, E., 2016. Automated Arabic text classification with P-Stemmer, machine learning, and a tailored news article taxonomy. *Journal of the Association for Information Science and Technology*. 67(11).

Elkhoury, R., Taghva, K., J., Coombs, 2005. Arabic Stemming without a Root Dictionary. In *ITCC'05, International Conference on Information Technology: Coding and Computing*. Vol. 2.

Mohod, S., Dhote, C., 2014. Feature Selection Technique for Text Document Classification: An Alternative Approach. *International Journal on Recent and Innovation Trends in Computing and Communication*. 2(9).

Gusev, I., Indenbom, E., Anastasyev, D., 2018. Improving Part-of-Speech Tagging via Multi-Task Learning and Character-Level Word Representations. In *Dialogue 2018, Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference.* (17).

Deshmukh, S., Shinde, G., 2016. Sentiment TFIDF Feature Selection Approach for Sentiment Analysis. *International Journal of Innovative Research in Computer and Communication Engineering*. 4(7).

Ojeda, T., Bilbro, R., Bengfort, B., 2018. *Applied Text Analysis with Python: Enabling Language-Aware Data Products with Machine Learning*. O'Reilly Media. 1st Edition.

Ghosh, S., Desarkar, M., 2018. Class Specific TF-IDF Boosting for Short-text Classification: Application to Short-texts Generated During Disasters. In *IW3C2, International World Wide Web Conference Committee*.

TfidfVectorizer. [cited 1-10-2019]; Available from: https://scikit learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

Kshirsagar, V., Awachate, B., 2016. Improved Twitter Sentiment Analysis Using NGram Feature Selection and Combinations. In *IJARCCE*, *International Journal of Advanced Research in Computer and Communication Engineering*. 5(9).

Bhayani, R., Huang, L., A., Go., 2009. *Twitter Sentiment Classification using Distant Supervision*. Stanford Digital Library Technologies Project.

Allison, B., Guthrie, D., Guthrie, L., 2006. Another Look at the Data Sparsity Problem. *International Conference on Text, Speech and Dialogue*.

Narayanan, V., Arora, I., Bhatia, A., 2013. Fast and Accurate Sentiment Classification Using an Enhanced Naïve Bayes Model. In *IDEAL, International Conference on Intelligent Data Engineering and Automated Learning.*

Cocea, M., Liu, H., 2017. Semi-random partitioning of data into training and test sets in granular computing context. *Granular Computing Journal*. 2(4).

Bazazeh, D., Shubair, R., 2016. Comparative Study of Machine Learning Algorithms for Breast Cancer Detection and Diagnosis. In *ICEDSA*, *The 5th International Conference on Electronic Devices, Systems and Applications*.

Xu, M., Wang, J., Zhang, J., Wang, H., 2006. Classification of Imbalanced Data by Using the SMOTE Algorithm and Locally Linear Embedding. In *IEEE*, *The 8th international Conference on Signal Processing.*

Pears, R., Finlay, J., Connor, A., 2014. Synthetic Minority Over-sampling Technique (SMOTE) for Predicting Software Build Outcomes. *The Twenty-Sixth International Conference on Software Engineering and Knowledge Engineering*.

Blagus, R., Lusa L., 2012. Evaluation of SMOTE for High-Dimensional Class-Imbalanced Microarray Data. In *IEEE*, *The 11th International Conference on Machine Learning and Applications*.

Kalajdziski, S., Lameski, P., Zdravevski, E., Kulakov, A., 2015. Transformation of Nominal Features into Numeric in Supervised Multi-Class Problems Based on the Weight of Evidence Parameter. *Proceedings of the Federated Conference on Computer Science and Information Systems*. Vol. 5.

Precision_Recall_Fscore_Support. [cited 1-10-2019]; Available from: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html

The Classification Report. [cited 29-09-2019]; Available from: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

Barendregt, L., Braak, C., 1986. Weighted Averaging of Species Indicator Values: Its Efficiency in Environmental Calibration. *Mathematical Biosciences Journal*. 78(1).

Weighted Averaging, *United States Environmental Protection Agency*. [cited 30-09-2019]; Available from: https://www.epa.gov/sites/production/files/2016-07/documents/weighted-averaging.pdf

Tagliaferri, R., Galdi, P., 2018. *Data Mining: Accuracy and Error Measures for Classification and Prediction*. Encyclopedia of Bioinformatics and Computational Biology: Elsevier, Vol. 1.

Grus, J., 2015. *Data Science from Scratch: First Principles with Python*. O'Reilly Media. USA, 1st edition.

Nkambou, R., Desmarais, M., Masthoff, J., Mobasher, B., 2012. User Modeling, Adaptation, and Personalization. In *UMAP*, *Proceedings the 20th International Conference*. *Springer*. Canada.

Franconeri, S., Bertini, E., Felix, C., 2018. Taking Word Clouds Apart: An Empirical Investigation of the Design Space for Keyword Summaries. *The Journal of IEEE Transactions on Visualization and Computer Graphics*. 24(1).

Jin, Y., 2017. Development of Word Cloud Generator Software Based on Python. *Procedia Engineering Journal*. 174(2017).