Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PREDICTING SARCASM AND POLARITY IN ARABIC TEXT AUTOMATICALLY: SUPERVISED MACHINE LEARNING APPROACH

MOHAMED ABDELRAZEK ABDELAAL[1] , MOHAMED ABDEL FATTAH[2], MONA MOHAMED ARAFA [3]

[1,2,3]Department of Information Systems, Faculty of Computer and Artificial intelligence, Benha University

13511, Egypt

E-mail: [1]mo.abdelrazeek@gmail.com, [2]mohamed.abdo@fci.bu.edu.eg, [2]mona.abdelmonem@fci.bu.edu.eg

ABSTRACT

Arabic text is one of the main challenges for machine learning and sentiment analysis to this day. In this paper, we introduce an Arabic text classifier that predicts both polarity and sarcasm. Six different supervised machine learning classification algorithms were used and gauged on our Arabic classifiers: Logistic Regression (LR), Multinomial Naïve Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM), Random Forest (RF) and K-Nearest Neighbor (KNN) along with different N-gram for tokenizing and TF-IDF for feature selection. SVM shows the best accuracy results (F1-score of 58.5%) when predicting polarity, while DT achieves the best accuracy results (F1-score of 64.4%) when predicting sarcasm. Previous sarcasm classification research achieved (F1-score of 46%) accuracy using BiLSTM on Arabic corpus.

Keywords: *Text Classification, Supervised Machine Learning, N-Gram, Sarcasm Detection*

1. INTRODUCTION

The Internet is the multidimensional platform that brings and has greatly shaped the nature of public opinion, which leads the different researchers to retrieve, process, and analyze these data effectively. The data are located in different resources such as social media networks and online news portals [1]. Social media are interactive computer technology applications that allows the creation, access, and exchange of user-generated content that generated and exchange by users, such as written posts and comments, ideas, careers, tweets, digital images, or videos.

The generated data that come through online virtual interactions makes social media the largest, and richest dynamic evidence base of human behavior and that help to understand society. Although social media and online news portals are the richest data sources, these data that available are not analyzed which has motivated innovative scientists and industry researchers to increasingly proposing new ways to collect and analyze this wealth of data automatically [2].

Data analysis is the process of cleaning and modelling data to discover hidden and useful information about it; Data mining is a data analysis technique that predicts and finds useful information contained in big data. One of the most known algorithms in data mining is sentiment analysis that analyzes the text using opinion mining algorithms and natural language processing algorithms to extract and study users opinion about the data [3].

Machine learning is a data analysis science that automatically studies, empowers and improve computers learning to intelligently act and predict like humans without being explicitly programmed. It is considered a branch and progress toward artificial intelligence. Machine learning has different learning methods or approaches to make the computer more intelligent such as supervised learning, semi supervised learning, unsupervised learning, and reinforcement learning. In this research, we focus on supervised learning that based on training the machine using labelled data inputs along with a known output to correctly predict new inputs that did not have a known output/result [4-8]. Machine learning and artificial intelligence algorithms are important because both are used to construct the big picture that helps decision-makers in their work to be empowered with valuable information and to form insights about the challenges and problems of the public.

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Sentiment analysis is a popular machine learning concept used to mine the text using artificial intelligence and NLP to determine and detect opinions about a given unstructured text. The sentiment is used to get the polarity of the input text and determine if it's negative, neutral, or positive and it's widely used nowadays in various fields where customer opinion is important. Sentiment analysis classifies the text into different classes to get useful insights and turn that text from unstructured text to structured one [9-13].

The purpose of this paper is to introduce a supervised opinion mining framework that correctly predicts both polarity and sarcasm for the Arabic text with high accuracy. In order to achieve that purpose, six different supervised machine learning classifiers with three n-grams are used.

The significant contributions of this paper are summarized as follow: The first contribution is building an efficient and effective opinion mining framework that predicts both polarity and sarcasm for the Arabic text. The second contribution is differentiating and gauging the prediction accuracy of six (SVM, DT, RF, LR, KNN, NB) different supervised machine learning algorithms. The third contribution is differentiating how the different n-grams can improve the overall accuracy.

The remainder of this paper structured as follows: In the second section, related works are presented. In the third section, the proposed framework is presented. In the fourth section, the results and discussion are presented before the conclusion in the fifth section.

2. RELATED WORKS

In [14] Abdullatif, et. al. introduced a systematic review that reviews 108 papers related to Arabic sentiment analysis researches from 2013 till 2018. The researcher's goals were to discuss the main issues affecting Arabic sentiment analysis by following a specific step that started with identifying four questions to know when this research published, who has published it, identifies the current state and the most effective techniques used, and what are the gaps or limitations in these studies. The next steps showed how the studies were selected from Scopus databases, how the authors include and exclude the used researches, and how the data were collected from these studies to identify other researchers used algorithms, datasets, classification level, how the features were selected and also how the preprocessing were done in all the studies. The authors conclude their study by highlighting the most used features such as n-grams and the most used algorithms in Arabic sentiment analysis such as Support Vector Machine and Naive Bayes algorithms.

In [15] Donia, et. al. introduced a comprehensive study among machine learning algorithms with different Arabic dialects and N-gram to determine the polarity of a given tweet. The authors used an Arabic tweets dataset collected from Twitter including 75774 negative tweets and 75774 positive tweets. The authors cleaned their data by applying the data preprocessing steps and to label their data as negative phrases or pwhositive phrases. The authors extracted features using used the different length N-gram then they apply ten different machine learning algorithms using Natural Language Tool Kit [16] and SKLearn library [17] to determine the polarity. The authors applied the 10-fold cross-validation to evaluate classifiers' performance. The authors conclude their study by highlighting classifiers higher-performing such as Ridge-Regression and Passive-Aggressive algorithms (99.96%).

In [18] Raddad, et. al. introduced a manually collected corpus that stores political comments and articles related to Arab's spring to analyze it and study the performance of different feature reductions along with machine learning algorithms. The authors showed how the data preprocessing is a must to prepare the data to be used in the classification and also explains different methods to tokenizing such as N-gram and bag of words. The authors used different machine learning algorithms to predict the political orientation of their corpus articles and comments. Algorithms such as Naive Bayes, support vector machine, random forest, sparse generative model, and also mixed classifiers (VOTE). The authors applied the 10-fold cross-validation to evaluate classifiers' performance and conclude their study by highlighting how the N-gram method showed superiority over the bag of words method. Also, they showed how the support vector machine classifier obtained the best results over all other algorithms.

In [19] Mohammed, et al. introduced a system that analyzes Arabic tweets and determines if this tweet is suspicious or not automatically and also monitor predict if a tweet writer is involved in illegal activity or a crime. The authors collected their data from Twitter and it's consisted of total 1555 tweets and these tweets include 826

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

suspicious and 729 are not. The authors labelled the collected tweets manually to build their dataset that used along with six different supervised machine learning algorithms. The authors analyzed the performance of the six algorithms using in terms of confusion matrices, execution time, and accuracy. These algorithms are long short-term memory networks, artificial neural networks, support vector machine, linear discriminant algorithm, k-nearest neighbors, and decision tree. The authors mentioned how their work is limited because they used a limited number of tweets. The authors conclude their study by highlighting how the support vector machine algorithm got the best performance with a mean accuracy of 86.72% followed by the decision tree algorithm.

In [20] Ibrahim and et al. Introduced a new dataset that was constructed from well-known Arabic sentiment analysis datasets SemEval [21] and ASTD [22]. The authors focused on the challenges that face Arabic sentiment analysis researches especially sarcasm detection. The authors train their model using a Bidirectional Long short-term memory BiLSTM algorithm and achieve an F1-score of 0.46 after utilising the embeddings provided by [23].

3. ArSarcasm DATASET

The proposed framework tested with ArSarcasm dataset that used for Arabic sarcasm detection and was created by Ibrahim, et al using previously available datasets ASTD [22], and SemEval-2017 [21] with the addition of two new columns called the dialect-label and the sarcasm-label. The dataset consists of more than 10000 tweets provided in CSV format and divided into 80/20 train/test and consists of these columns as shown in the below Table 1.

Table 1: ArSarcasm dataset

field	Description
Sarcasm	True or False (Boolean) that indicates if the tweet is labelled sarcastic or not.
Tweet	text from the original tweet.
Source	the tweet original source ASTD or SemEval
Orignal Sentiment	the original sentiment from ASTD or SemEval (negative, positive, neutral).
Dialect	the dialect used in the tweet (Egypt, MSA, Maghreb, gulf, levant).

The next visualizations show how the sarcasm, dialect, and sentiment columns are distributed, as shown in Figure 1, 2, 3, 4 below.

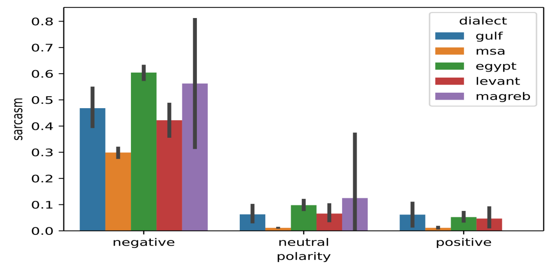

Figure 1: Dataset training data visualizations Sarcasm, dialects polarity

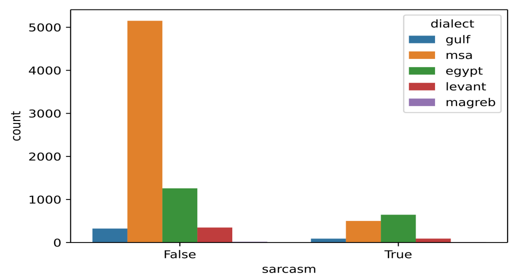

Figure 2: Dataset training data visualizations Dialect's sarcasm

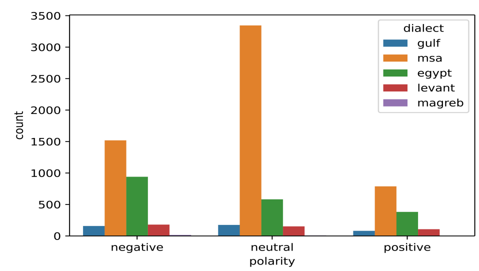

Figure 3: Dataset training data visualizations Dialect's polarity

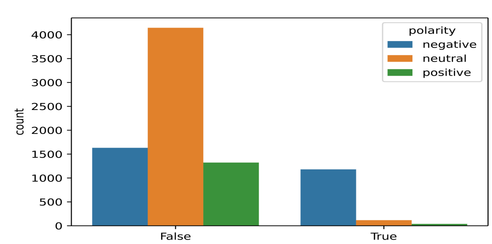

Figure 4: Dataset training data visualizations Sarcasm polarity

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. SUPERVISED MACHINE LEARNING CLASSIFIERS

We used, gauged, and considered six different sentiment machine learning classification algorithms: Logistic Regression, Multinomial Naïve Bayes, Decision Tree (DT), Support Vector Machines (SVMs), K-Nearest Neighbor (KNN), and Naive Bayes. Our experiments use the scikit-learn the free software machine learning library, version 0.23.2 [17]; to implement and measure/evaluate the applicability for the below classifiers and only use Naïve Bayes implementation from The Natural Language Toolkit (NLTK) [16], version 3.5.

4.1 Logistic regression

Logistic regression is a machine learning algorithm used for supervised classification to predict a target variable probability that has only two possible classes. The target variable is a binary on nature, which means representing the variable as either 0 or 1 (equal to no/failure/false/lose or to yes/success/true/win). A logistic regression model in Mathematics predicts P(X=1) as a function of Y. Logistic regression is one the most known machine learning classifiers and also a simple one used for different classification tasks such as disease prediction, spam detection, gender detection etc. Logistic regression follows the sigmoid function like Equation (1) that maps any real value into another value between 1 and 0, and these values making an S-shaped curve.

$$S(a) = \frac{1}{1+e^{-a}} \tag{1}$$

where: S (a) is the probability estimate that produces output between 0 and 1, a is the what we need to predict and the input to the function e.g. mx + b, and e is the base of natural log.

4.2 Naive bayes

A Naïve Bayes classifier is a simple probabilistic classifier based on probability theorem (Bayes) that includes strong features' independence assumptions. Our Naïve Bayes classifier assigns a given tweet (t) the class c* as shown in Equation (2), then applying Bayes' rule as shown in Equation (3).

$$C^* = p(C_k|x_1,\dots,x_n) = Argmax_c P(n|p); c \in \{Positive, Negative\}||\{True, False\} \tag{2}$$

$$P(n|p) = \frac{P(n)P(p|n)}{P(p)} \tag{3}$$

P(p) plays no role in assigning c*. To estimate the term P (n | p), Naïve Bayes calculate the estimate by assuming all the f_i's are conditionally independent given p's class. Term n_i (t) is the presence of term i in tweet t(0or1).

$$P_{NB}(n|p) = \frac{P(n)\left(\prod_{i=1}^{m} P(f_i|c)^{n_i(t)}\right)}{P(p)} \tag{4}$$

4.3 Decision tree

Decision tree classifier is an intelligent and simple machine learning predictive model approach that is represented as a tree to go from an observation about an item to a conclusion about the target value for the selected item. The decision tree is a combination of computational and mathematical techniques to aid the categorization, description and generalization of a given set of data. Decision tree represented as leaves and branches where the class labels represented as leaves and the conjunctions of features that lead to those class labels is represented as a branch. The decision tree can be used to explicitly and visually represent decision making and decisions. The goal of using a decision tree is to create a model that correctly predicts the value of a target variable based on different input variables. Decision tree data comes in records of the form of Equation (5)

$$(f,T) = (f_1, f_2, f_3, \dots, f_k, T) \tag{5}$$

The dependent variable, T, is the target variable that we are trying to generalize, classify, or understand. The vector (f) is composed of the features, f_1, f_2, f_3, etc., that are used for that task. The decision tree has different metrics to measure the quality of a split such as the Gini impurity; to compute Gini impurity for a set of items with T classes; equation 6 show how to apply the Gini impurity. Suppose n\in {1,2 ,3, ..., T} and let p_n be the fraction of items labelled with class n in the set. The dependent variable, T, is the target variable that we are trying to generalize, classify, or understand. The vector (f) is composed of the features, f_1, f_2, f_3, etc., that are used for that task. The decision tree has different metrics to measure the quality of a split such as the Gini impurity; to compute Gini impurity for a set of items with T classes; equation 6 show how to apply the Gini impurity. Suppose n ∈ {1, 2, 3, ..., T} and let p_n be the fraction of items labelled with class n in the set.

$$N_G(p) = \sum_{n=1}^{T}\left(p_n \sum_{k\neq n} p_k\right) = \sum_{n=1}^{T} p_n(1-p_n) = \sum_{n=1}^{T}(p_n - p_n^2) = \sum_{n=1}^{T} p_n - \sum_{n=1}^{T} p_n^2 = 1 - \sum_{n=1}^{T} p_n^2 \quad (6)$$

4.4 Support vector machine

SVMs separates two categories or classes of data by identifying decision surface (or hyperplane). The selected hyperplane generates the largest margin or separation between the two classes; therefore, it is a large margin classifier. Suppose we have n tweets to be grouped or categorized. Our set S of tweets is represented as Equation (7) where f_i represents the features of the tweet; and s_i represents the categorization of that tweet, either a positive or a negative for polarity detection and a true or a false for sarcasm detection; set of weights w(orw_i) one for each feature f_i, whose linear combination predicts the value of s_i.

$$S = \{(f_1, s_1), (f_2, s_2), (f_3, s_3) \ldots (f_n, s_n)\} \quad (7)$$

We need to minimize the ||w|| (the space between the two parallel hyperplanes) to maximize the margin (‘street width’) by our two classes hyperplane H_1, H_2 as Equation (8):

$$w \cdot f_i + b \geq +1 when s_i = +1$$

$$w \cdot f_i + b \leq -1 when s_i = -1 \quad (8)$$

The quadratic optimization problem is to minimize ||w||, so we rewrite the equations as Equation (9). This will identify the largest margin between our negative and positive tweets when detecting polarity or between our true and false tweets when detecting sarcasm.

$$s_i(w \cdot f_i) - b = 1 or [s_i(w \cdot f_i) - b] - 1 = 0 \quad (9)$$

4.5 K-Nearest Neighbor

The KNN algorithm is used to classify a set of inputs by assigning a class membership for each input. KNN classify an object by the vote of the plurality of its neighbors, with the object being assigned to the most common class among its k nearest neighbors. KNN k is typically a positive small integer; suppose k=1, then the object is simply assigned to the class of that single nearest neighbor. KNN simply uses distance functions to measure each object similarity with its neighbors and is considered a non-parametric technique. The distance functions that are used with KNN are different such as integer-valued vectors (Minkowski, WMinkowski, Manhattan, Euclidean, Chebyshev, Seuclidean, Mahalanobis), and real-valued vectors (Hamming, Braycurtis, Canberra). Manhattan, Minkowski, Euclidean, and Hamming are represented as Equation (10), Equation (11), Equation (12), and Equation (13)

$$D_{Manhattan} = \sum_{i=1}^{K} |x_i - y_i| \quad (10)$$

$$D_{Minkowski} = \left(\sum_{i=1}^{K} (|x_i - y_i|)^q\right)^{1/q} \quad (11)$$

$$D_{Euclidean} = \sqrt{\sum_{i=1}^{K} (x_i - y_i)^2} \quad (12)$$

$$D_{Hamming} = \sum_{i=1}^{K} |x_i - y_i| \quad (13)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$x \neq y => D = 1$$

$$x = y => D = 0$$

5. EXPERIMENTS

The proposed framework is Arabic classifier that predicts both polarity and sarcasm, like any other text-mining classifier, it follows different steps: pre-processing, feature selection, training, and testing.

Preprocessing: The preprocessing is a critical and useful step to clean and prepare the data for the classification step. The preprocessing includes different steps such as:

Remove stop words: such as at, is, the, why, which.

Stemming: Figuring out the root or stem of words.

Tokenization: Tokenization is to break the text into a set of tokens for computing feature vectors, and it has different types: Bag of words (BOW) and N-gram (unigram, bigram, trigram).

Feature Selection: Is the process used to select and reduce the best features to reduce the overall learning cost by eliminating as much redundant and unrelated information as possible.

Training and Testing: Large dataset need to be divided into a separate train and test sets. The larger we make our train and test set, the better module we'll be able to trust. If the dataset we have is very large, sounds good, but we might still need to hold out 10-20% for the tests. If the dataset we have is very small, then we might need to apply something like cross-validation.

The **n-gram** is n items contiguous sequence from a given sample of a collected tweet or text. The n-gram normally use the Latin numerical prefix, an n-gram of size 1 is referred to as a "1-gram" or "unigram"; size 2 is a "2-gram" or "bigram"; size 3 is a "3-gram" or "trigram"; size 4 is a "4-gram" or "four-gram", and so on. Below table 1 show examples for different n-grams for tweets.

Table 2: N-gram Tweets sample

Sample Tweet	1-gram sequence	Sample Tweet	1-gram sequence
مصر لن تركع (Egypt will not kneel)	مصر، لن، تركع Egypt, will, not, kneel	مصر لن، لن تركع، تركع... Egypt will, will not, not kneel, kneel...	مصر لن تركع Egypt will not, will not kneel, not kneel ….
مصر لها شعب يحميها، رب أجعل هذا البلد آمنا مطمئنا. (Egypt has a people that protects it, may God make this country safe and secure.)	مصر، لها، شعب، يحميها، رب، اجعل، هذا، البلد، امنا، مطمئنا…. Egypt, has, a, people, that, protects, it, may, God, make, this, country, safe, and, secure,	مصر لها، لها شعب، شعب يحميها، يحميها رب، رب اجعل، اجعل هذا …. Egypt has, has a, a people, people that, that protects, ….	مصر لها شعب، لها شعب يحميها، يحميها رب أجعل، رب اجعل هذا، اجعل هذا البلد، هذا البلد أمنا،… Egypt has a, has a people, a people that, people that protects, …..
مصر توقع اتفاقيه مبادله العملات مع الصين مليار 26بقيمه دولار (Egypt signs a currency swap agreement with China worth $26 billion)	مصر، توقع، اتفاقية، مبادلة، العملات، مع، الصين، بقيمه، ٢٦، مليار، دولار،…. Egypt, signs, a, currency, swap, agreement, China, ….	مصر توقع، توقع اتفاقية، اتفاقية مبادلة، مبادلة العملات، العملات بقيمة، بقيمة ٢٦، … Egypt signs, signs a, a currency, currency swap…	مصر توقع اتفاقية، توقع اتفاقية مبادلة، اتفاقية مبادلة العملات،……. Egypt signs a, signs a currency, a currency …..
هاتف جوجل يحتوي علي أفضل كاميرا صُنعت علي الإطلاق (Google phone has the best camera ever made)	هاتف، جوجل، يحتوي، علي، أفضل، كاميرا، ….. Google, phone, has, the, best, camera, ….	هاتف جوجل، جوجل يحتوي، يحتوي علي، علي أفضل، …. Google phone, phone has, has the, the best, best ….	هاتف جوجل يحتوي، جوجل يحتوي علي، يحتوي أفضل كاميرا،…. Google phone has, phone has the, has the best, ….

In Table 2, we show how different n-grams work, but it is important to know that machine learning has different preprocessing steps where stop words are removed, each word goes back to its root, etc.; And when we get to the tokenization step where the n-gram occurs; we will not have any stop words (at, is, the, why, which) to be included in the n-grams. Example: "Egypt signs a currency swap agreement with China"; This sentence will be "Egypt sign currency swap agreement China"; Then the n-gram starts its work;

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

We did not apply any preprocessing to the sentence to simply explain how the n-gram works.

5.1 Arabic polarity and sarcasm detector

The proposed framework was gauged using six different text-mining classifiers (SVM, DT, RF, LR, KNN, and Multinomial Naive Bayes) mentioned in the fourth section and different validation techniques. The proposed framework has two modules: Arabic Polarity and Sarcasm modules, the first module detects the polarity and the second detect the sarcasm; the first classifies the input tweet as positive, negative, or neutral and the second classifies the input tweet as a True or False. Each module differentiates the accuracy results for the six classifiers along with unigram or bigram or trigram with TF-IDF. Both modules follow the next Algorithm 1 and the Equation (14) as shown in Figure 5 below:

(unigram || bigram || trigram) && (TF-IDF) && (SVM || DT || LR || KNN || NB || RF) **(14)**

Algorithm-1: The proposed framework

I. Preprocessing
   I. Removing stop words
   II. Stemming
   III. Tokenizing
II. Feature Selection
   I. TF-IDF
III. Training
IV. Testing

In order to compare the accuracy for the different grams and differentiate between learning costs when using different algorithms configurations, we have got 18 different accuracy results (Precision, Recall, F1-Measure) for each module. In order to understand more about the proposed framework; algorithm-1 as mentioned above and the flowchart as shown in Figure 5 above will clarify how the proposed framework works.

6. RESULTS AND DISCUSSIONS

Using our six classifiers mentioned in Equation (14), we have applied the different n-gram with TF-IDF, Table 3 shows the different results for the used classifiers: SVM, DT, LR, KNN, NB, and RF. Using our six classifiers, Section 6.2 that discuss polarity results shows that:

- Support SVM with (bigram, unigram, trigram)-TF-IDF, and LR with (unigram, bigram)-TF-IDF performs well.
- RF with (unigram, bigram, trigram)-TF-IDF perform adequately for predicting the Arabic text polarity.

While Section 6.3 that discuss sarcasm results shows that:

- SVM with (trigram, bigram, unigram)-TF-IDF
- RF with (unigram, bigram, trigram)-TF-IDF, KNN with bigram-TF-IDF.
- LR with unigram-TF-IDF performs well
- NB with unigram-TF-IDF perform adequately for prediction the Arabic text sarcasm.

Generally, on average unigram as shown in Figure 6 achieved the best results when it was used with the different classifiers and TF-IDF.

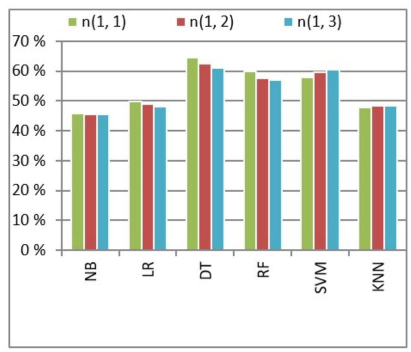

Figure 6: N-grams F1-Measure results

6.1 Proposed work VS Sarcasm detection baseline in [20]

The Arabic sarcasm detector as shown in Section 6.2 got better results (F1-score of 64%) than the Sarcasm Detection Baseline System (F1-score of 46%) mentioned in [20] in previous works Section 3. Table 3 below show a comparison between our proposed work and the work introduced in [20].

Table 3: Proposed work vs Sarcasm detection baseline

Model Name	Arabic Sarcasm detector	Sarcasm Detection Baseline System
Technology	Machine Learning	Deep Learning
Algorithm	SVM, DT, LR, KNN, NB, and RF	BiLSTM
Learning Costs	Low	High
F1-score	F1-score of 64%	F1-score of 46%

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6.2 Polarity results

This module is used to detect the polarity (neutral, positive, or negative) of the Arabic text. Tables 4, 5, 6, and 7 below show the Accuracy and F1-Measure results. Generally, on average SVM as shown in Figure 7 and Figure 8 achieved the best results followed by in order RF, LR, DT, KNN, NB.

Table 4: Polarity Precision Results

Algorithm	Precision		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	0.77	0.77	0.77
LR	0.66	0.66	0.66
DT	0.53	0.50	0.49
RF	0.61	0.59	0.58
SVM	0.64	0.64	0.64
KNN	0.63	0.61	0.61

Table 5: Polarity Recall Results

Algorithm	Recall		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	0.77	0.46	0.42
LR	0.66	0.54	0.53
DT	0.49	0.51	0.50
RF	0.58	0.55	0.54
SVM	0.64	0.57	0.57
KNN	0.61	0.48	0.47

Table 6: Polarity Accuracy Results

Algorithm	Accuracy		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	62.227	59.384	58.673
LR	**65.498**	**65.024**	**64.739**
DT	57.583	55.118	54.787
RF	**63.507**	**62.986**	**62.275**
SVM	**65.829**	**65.924**	**65.545**
KNN	60.616	59.810	60.379

Table 7: Polarity F1-Measure Results

Algorithm	F1-Measure		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	58.673	43.367	39.012
LR	**64.739**	55.830	53.448
DT	54.787	51.881	49.669
RF	**62.275**	56.033	55.318
SVM	**65.545**	**58.353**	**58.587**
KNN	60.379	48.404	47.882

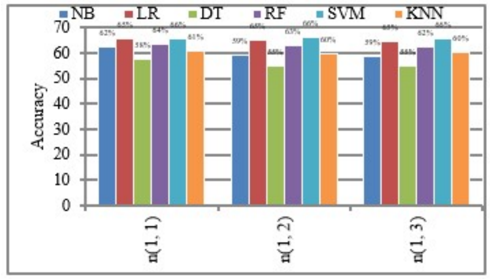

Figure 7: Polarity Accuracy results

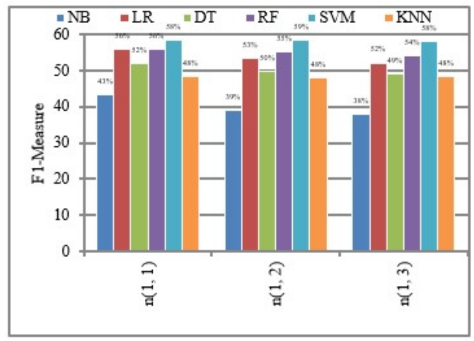

Figure 8: Polarity F1-Measure results

6.3 Sarcasm results

This module is used to detect the sarcasm (True or False) of the Arabic text. Tables 8, 9, 10, and 11below show the Accuracy and F1-Measure results. Generally, on average DT as shown in Figure 9 and Figure 10 achieved the best results followed by in order SVM, RF, LR, KNN, NB.

Table 8: Sarcasm Precision Results

Algorithm	Precision		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	0.42	0.42	0.42
LR	0.75	0.75	0.79
DT	0.66	0.64	0.62
RF	0.74	0.73	0.73
SVM	0.75	0.74	0.74
KNN	0.78	0.84	0.80

Table 9: Sarcasm Recall Results

Algorithm	Recall		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	0.50	0.50	0.50
LR	0.52	0.52	0.51
DT	0.63	0.62	0.60
RF	0.58	0.56	0.56
SVM	0.57	0.58	0.58
KNN	0.51	0.51	0.51

Table 10: Sarcasm Accuracy Results

Algorithm	Accuracy		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	83.697	83.649	83.649
LR	**84.028**	83.934	83.934
DT	82.133	81.422	80.427
RF	**84.739**	**84.455**	**84.502**
SVM	**84.739**	**84.834**	**84.929**
KNN	83.886	**84.028**	83.981

Table 11: Sarcasm F1-Measure Results

Algorithm	F1-Measure		
	n (1, 1)	n (1, 2)	n (1, 3)
NB	45.849	45.548	45.548
LR	49.961	48.911	48.133
DT	**64.408**	**62.643**	**61.216**
RF	**59.794**	57.533	57.041
SVM	57.963	59.727	**60.449**
KNN	47.848	48.439	48.417

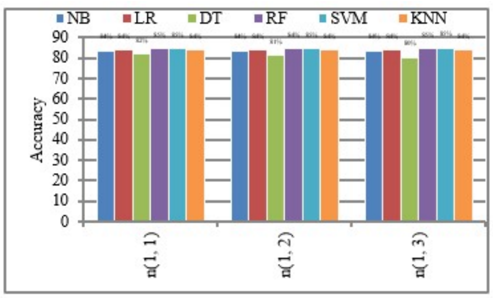

Figure 9: Sarcasm Accuracy results

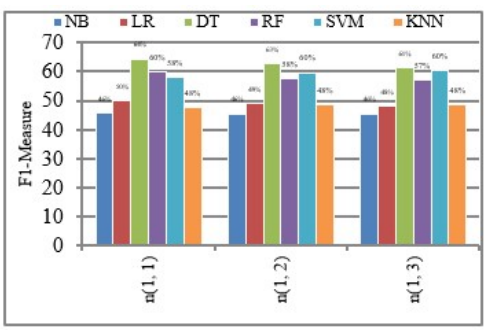

Figure 10: Sarcasm F1-Measure results

7. CONCLUSION

In this paper, we introduce an Arabic text classifier that predicts both polarity and sarcasm. Six different supervised machine learning classification algorithms were used and gauged on our Arabic classifier: Logistic Regression (LR), Multinomial Naïve Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM), and K-Nearest Neighbor (KNN) along with different N-gram for tokenizing and TF-IDF for feature selection. The proposed work shows how the unigram got the best results when mining the Arabic text. SVM shows the best accuracy results (F1-score of 58.5%) when predicting polarity, while DT achieves the best accuracy results (F1-score of 64.4%) when predicting sarcasm. Previous sarcasm classification research achieved (F1-score of 46%) accuracy using BiLSTM on Arabic corpus. The results show how the proposed work correctly predicts both polarity and sarcasm for the Arabic text with high accuracy.

REFERENCES:

[1] G. Fortino, R. Buyya, M. Chen and F. Herrera, "Special Issue on Methods and Infrastructures for Data Mining at the Edge of Internet of Things," in IEEE Internet of Things Journal, vol. 8, no. 13, pp. 10220-10221, 1 July1, 2021, doi: 10.1109/JIOT.2021.3075304.

[2] H. N. Desai and R. Patel, "A Study of Data Mining Methods for Prediction of Personality Traits," 2020 International Conference on Smart Electronics and Communication (ICOSEC), 2020, pp. 58-64, doi: 10.1109/ICOSEC49089.2020.9215379.

[3] U. Narayanan, V. Paul and S. Joseph, "Different analytical techniques for big data analysis: A review," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), 2017, pp. 372-382, doi: 10.1109/ICECDS.2017.8390139.

[4] S. Chowdhury and M. P. Schoen, "Research Paper Classification using Supervised Machine Learning Techniques," 2020 Intermountain Engineering, Technology and Computing (IETC), 2020, pp. 1-6, doi: 10.1109/IETC47856.2020.9249211.

[5] C. M. Suneera and J. Prakash, "Performance Analysis of Machine Learning and Deep Learning Models for Text Classification," 2020 IEEE 17th India Council International Conference (INDICON), 2020, pp. 1-6, doi: 10.1109/INDICON49873.2020.9342208.

[6] M. Ismail B., M. Alam, M. Tahernezhadi, H. K. Vege and P. Rajesh, "A Machine Learning Classification Technique for Predicting Prostate Cancer," 2020 IEEE International Conference on Electro Information Technology (EIT), 2020, pp. 228-232, doi: 10.1109/EIT48999.2020.9208240.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[7] E. Nazarenko, V. Varkentin and A. Minbaleev, "Application for Traffic Classification Using Machine Learning Algorithms," 2020 International Conference Quality Management, Transport and Information Security, Information Technologies (IT&QM&IS), 2020, pp. 269-273, doi: 10.1109/ITQMIS51053.2020.9322895.

[8] S. Joshi and E. Abdelfattah, "Efficiency of Different Machine Learning Algorithms on the Multivariate Classification of IoT Botnet Attacks," 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2020, pp. 0517-0521, doi: 10.1109/UEMCON51285.2020.9298095.

[9] J. F. Raisa, M. Ulfat, A. Al Mueed and S. M. S. Reza, "A Review on Twitter Sentiment Analysis Approaches," 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), 2021, pp. 375-379, doi: 10.1109/ICICT4SD50815.2021.9396915.

[10] Y. Zhang, J. Sun, L. Meng and Y. Liu, "Sentiment Analysis of E-commerce Text Reviews Based on Sentiment Dictionary," 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), 2020, pp. 1346-1350, doi: 10.1109/ICAICA50127.2020.9182441.

[11] T.Vijay, A. Chawla, B. Dhanka and P. Karmakar, "Sentiment Analysis on COVID-19 Twitter Data," 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 2020, pp. 1-7, doi: 10.1109/ICRAIE51050.2020.9358301.

[12] M. Gupta, A. Mishra, G. Manral and G. Ansari, "Aspect-category based Sentiment Analysis on Dynamic Reviews," 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA), 2020, pp. 492-496, doi: 10.1109/ICCCA49541.2020.9250914.

[13] Al Shamsi, Arwa & Bayari, Reem & Salloum, Said. (2021). Sentiment Analysis in English Texts. Advances in Science Technology and Engineering Systems Journal. 5. 1683-1689. 10.25046/aj0506200.

[14] Abdullatif Ghallab, Abdulqader Mohsen, Yousef Ali, "Arabic Sentiment Analysis: A Systematic Literature Review", Applied Computational Intelligence and Soft Computing, vol. 2020, Article ID 7403128, 21 pages, 2020.

[15] Chatterjee, Soumick & Nath, Asoke. (2017). Auto-Explore the Web – Web Crawler. International Journal of Innovative Research in Computer and Communication Engineering. 5. 6607-6618. 10.15680/IJIRCCE.2017.0504006.

[16] Loper, Edward & Bird, Steven. (2002). NLTK: the Natural Language Toolkit. CoRR. cs.CL/0205028. 10.3115/1118108.1118117.

[17] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[18] Raddad Abooraig, Shadi Al-Zu'bi, Tarek Kanan, Bilal Hawashin, Mahmoud Al Ayoub, Ismail Hmeidi, Automatic categorization of Arabic articles based on their political orientation, Digital Investigation, Volume 25, Pages 24-41, ISSN 1742-2876, 2018.

[19] AlGhamdi, M.A., Khan, M.A. Intelligent Analysis of Arabic Tweets for Detection of Suspicious Messages. Arab J Sci Eng 45, 6021–6032 (2020).

[20] Abu Farha, Ibrahim & Magdy, Walid. (2020). From Arabic Sentiment Analysis to Sarcasm Detection: The ArSarcasm Dataset.

[21] Rosenthal, S., Farra, N., and Nakov, P. (2017). SemEval- 2017 task 4: Sentiment analysis in Twitter. In Proceed- ings of the 11th International Workshop on Semantic Evaluation, SemEval ’17, Vancouver, Canada, August. Association for Computational Linguistics.

[22] Nabil, M., Aly, M., and Atiya, A. (2015). Astd: Arabic sentiment tweets dataset. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 2515–2519.

[23] Abu Farha, I. and Magdy, W. (2019). Mazajak: An online Arabic sentiment analyser. In Proceedings of the Fourth Arabic Natural Language Processing Workshop, pages 192–198, Florence, Italy, August. Association for Com- putational Linguistics.

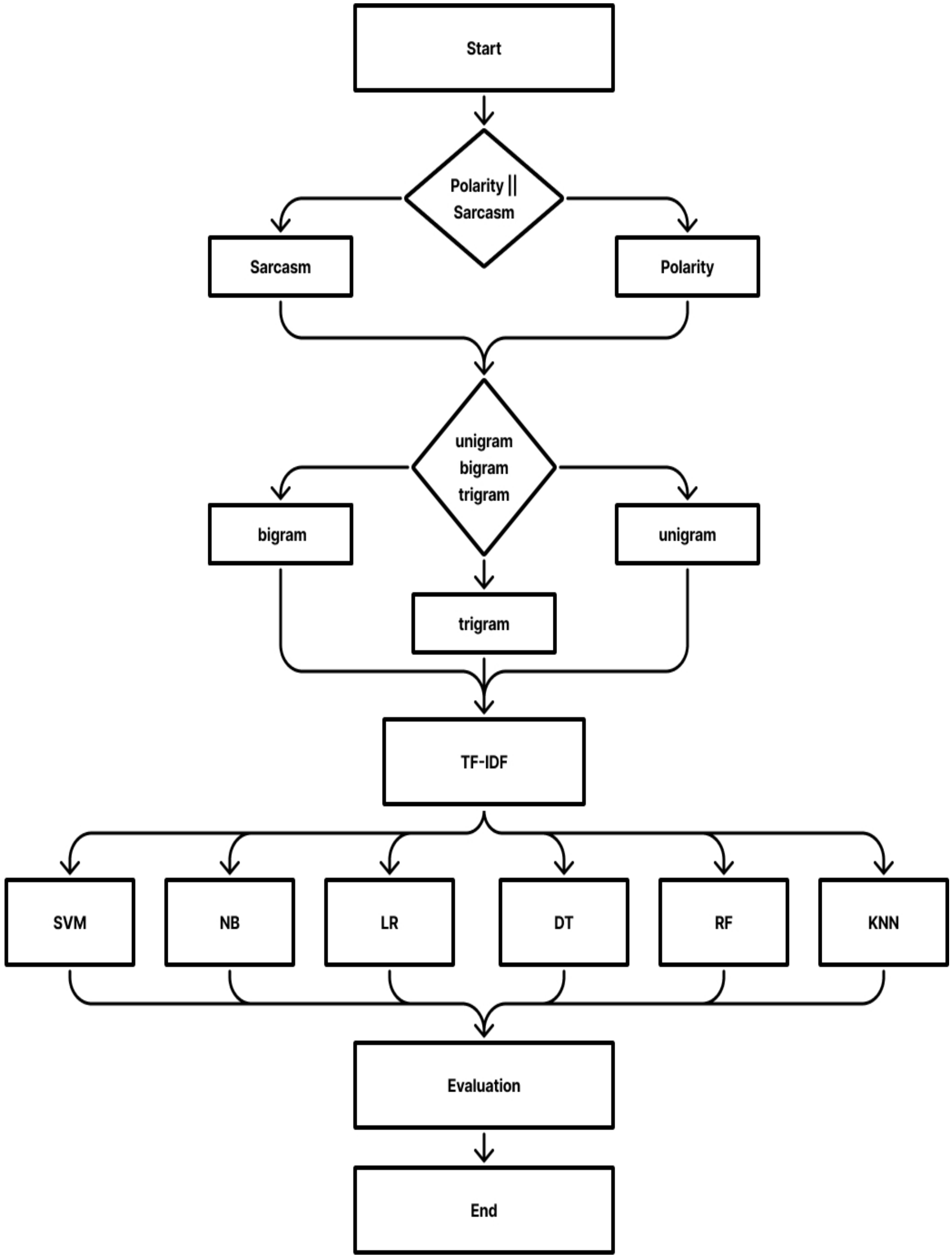

Figure 5: The proposed framework