Ingénierie des Systèmes d'Information
Vol. 28, No. 6, December, 2023, pp. 1435-1443
Journal homepage: http://iieta.org/journals/isi

Sentiment Analysis of Arabic Tweets on Online Learning During the COVID-19 Pandemic: A Machine Learning and LSTM Approach

Shahd Ebrahim Alqaan, Ali Mustafa Qamar*

Department of Computer Science, College of Computer, Qassim University, Buraydah 51452, Saudi Arabia

Corresponding Author Email: al.khan@qu.edu.sa

https://doi.org/10.18280/isi.280601

Received: 20 September 2023
Revised: 25 November 2023
Accepted: 4 December 2023
Available online: 23 December 2023

Keywords:
ARABIC tweets, COVID-19, deep learning (DL), long short-term memory (LSTM), machine learning (ML), online learning, opinion mining, sentiment analysis

ABSTRACT

In response to the unprecedented shift towards online learning during the COVID-19 pandemic, this study presents an innovative analysis of sentiments expressed in Arabic tweets. Utilizing a dataset comprising approximately 100,000 posts from the social media platform X (formerly known as Twitter), collected between 2020 and 2021, sentiments are explored surrounding two prevalent online learning hashtags: منصة_مدرستي (Madrasti platform, translating to 'My School platform') and التعليم_عن_بعد (Distance Learning). These hashtags were predominantly used by educational professionals, students, and teachers, reflecting their experiences with online education and the Madrasti platform. The dataset was initially imbalanced, with a significant skew towards negative sentiments. To address this imbalance and enhance the reliability of the analysis, Synthetic Minority Over-sampling Technique (SMOTE) and random under-sampling methods were employed. The balanced dataset was then subjected to sentiment analysis using different supervised machine learning (ML) algorithms, including Support Vector Machine (SVM), K nearest neighbor (KNN), and Random Forest, along with the long short-term memory (LSTM) as a deep learning (DL) algorithm. The experiments are conducted using a 10-fold cross-validation approach. The results showed a marked improvement in the precision, recall, and F-measure of the ML algorithms when applied to the balanced dataset, as opposed to the original imbalanced one. The performance of traditional ML classifiers was much better than that observed for LSTM. This research offers a detailed analysis of sentiments related to online learning during the pandemic and critically assesses different ML techniques in processing Arabic language data. The study's innovative approach to balancing the dataset and its extensive evaluation of different algorithms contribute significantly to sentiment analysis and opinion mining, particularly in the context of online education during a global health crisis.

1. INTRODUCTION

Since education is the foundation of any society and contributes to the development of future generations, it is essential. Despite the coronavirus's proliferation in 2021, Saudi Arabia opted to prioritize the continuity of the educational process rather than delaying it through the adoption of online learning. Online learning, which encompasses both synchronous and asynchronous communication, is often facilitated through various learning management systems (LMS) [1].

Administrators and educators must comprehend people's viewpoints through sentiment analysis to make online learning as successful as traditional learning by viewing students' opinions and the challenges they have been going through that make online learning harder for them and trying to make it easier and more beneficial for students in the future. Furthermore, they may learn about the benefits and drawbacks of following this approach.

Sentiment analysis aims to categorize opinions, feelings, and assessments. One of sentiment analysis's fundamental principles is recognizing the entire text's polarity. Sentiment analysis helps determine whether a given text carries a positive, negative, or neutral sentiment. This analysis often unveils people's opinions about a product, enhancing its quality and consistency [2].

People use social media sites such as X, previously known as Twitter, to discuss various life aspects [3]. Private companies and governmental organizations are two groups that have actively monitored and taken an interest in these communications. Nevertheless, scholars from diverse academic disciplines have recognized Twitter as a valuable knowledge resource that can be harnessed to monitor the evolution of any field.

The world shifted to online education after the advent of COVID-19. During the early stages of the pandemic, Saudi Arabia also chose online learning. In this study, we analyzed public opinion about online education using data obtained from Twitter. Deep learning (DL), which is a branch of machine learning (ML), has been incredibly successful in a variety of applications [4].

Alqaan and Qamar [5] have previously worked on Arabic

tweets related to online learning. However, only traditional ML algorithms were applied to an unbalanced dataset. Because of employing an unbalanced dataset, the obtained results were not good. The current research tries to bridge this gap, and the dataset introduced by Alqaan and Qamar [5] is balanced using the Synthetic Minority Over-sampling Technique (SMOTE) and the random under-sampling approach. This process eventually improved the results. Furthermore, we also utilize deep learning to create an effective sentiment analysis model, specifically using the long short-term memory (LSTM) approach.

The remaining paper is organized as follows: The state-of-the-art research comprising recent relevant research is provided in Section 2, and the methodology is discussed in the next section. The methodology includes the details about the data set along with a brief description of data cleaning and annotation tasks. Various preprocessing steps are detailed in Section 3, followed by a brief overview of the classification algorithms. The experiments, along with the results, are provided in Section 4. The impact of SMOTE as an over-sampling and an under-sampling technique is discussed in detail. Furthermore, the results obtained with LSTM are also provided. The paper concludes with Section 5, which reflects on the study's findings and explores potential avenues for future research.

2. RELATED WORKS

Sentiment analysis is a crucial domain within natural language processing (NLP) that focuses on identifying the subjectivity within the text by extracting and categorizing opinions and sentiments. It delves into understanding individuals' perspectives, attitudes, and emotions towards various products and services. Sentiment analysis finds applications in diverse fields, including business, politics, and finance, among others.

Dhawan et al. [6] proposed a sentiment polarity-based sentiment analysis mechanism for a Twitter dataset in online social networks. They conducted analytical work under the proposed method, measured the polarity of each tweet, and distinguished whether it was positive or negative. In comparison, our research also focuses on data obtained from Twitter. However, we consider three classes.

Wongkar and Angdresey [7] collected data from Twitter using a crawler and compared three different methods: Naïve Bayes (NB), Support Vector Machine (SVM), and K nearest neighbors (KNN). The accuracy for NB, SVM, and KNN was 75.58%, 63.99%, and 73.34%, respectively. Likewise, the present study utilizes KNN, Random Forest (RF), and SVM yet focuses on different aspects of sentiment analysis.

Ullah et al. [8] applied sentiment analysis to the airline industry using text and emoticons and used comments available on Twitter. Their approach worked with multiple features, including the Bag of Words model, and they employed a range of ML and DL algorithms. The outcomes with ML algorithms were as follows: 78% accuracy for logistic regression, 52% for NB, 76% for RF, and 78% for SVM. In contrast, DL algorithms demonstrated better performance, with LSTM achieving 89% accuracy and Convolutional Neural Network (CNN) getting 81%. In contrast, the research presented here has achieved an F-measure of 86% using RF.

Almalki's [9] study marks a significant advancement in sentiment analysis, specifically tailored for the distance learning domain within Saudi Arabia. Utilizing Apache Spark and the Twitter API, coupled with a comprehensive preprocessing methodology, Almalki's work demonstrates the capacity to extract meaningful insights from Arabic tweets about distance learning. His dataset, comprising 14,000 tweets, surpasses the volume utilized in the current research. The model exhibited remarkable efficiency, as evidenced by the performance metrics on the test data. Logistic Regression (LR) achieved an impressive accuracy of 91%, an F1 score of 90%, a precision of 90%, and a recall of 89%. However, SVM's performance was relatively lower, with an accuracy of 69%, aligning with the present study's findings, where RF surpassed SVM in effectiveness.

In another related study, Al-Harbi and Emam [3] focused on sentiment analysis within the context of Arabic text, particularly from the Saudi dialect. Their research involved an online dataset extracted from Twitter, with experiments conducted using the RapidMiner tool. Three supervised ML models were employed: KNN, NB, and SVM. These models were tested under three distinct preprocessing conditions: without stemming, using an Arabic light stemmer, and employing the Arabic stemmer. The results indicated that KNN and SVM outperformed NB. Notably, KNN achieved the highest accuracy of 37.07% without stemming, closely followed by SVM with an Arabic stemmer at an accuracy of 36.96%. The KNN classifier without stemming also recorded the best recall and precision at 84.25% and 44%, respectively.

The contribution of Aljabri et al. [10] to sentiment analysis in the context of online or distance learning in Saudi Arabia during the COVID-19 pandemic is noteworthy. Their approach differed by categorizing tweets into positive or negative classes, omitting neutral sentiments. The study utilized two datasets: a labeled one containing 5,096 tweets, reduced to 3,480 post-random under-sampling, and an unlabeled dataset comprising 9,160 tweets. The latter was primarily used to ascertain tweets pertinent to different educational stages and to identify the most frequent words in positive and negative classes. The algorithms employed included KNN, LR, NB, RF, SVM, and XGBoost, with logistic regression achieving a notable accuracy of 0.899, paralleling Almalki's findings [9]. Term Frequency – Inverse Document Frequency (TF-IDF) was utilized for feature extraction.

In a different domain, Yu and Zhang [11] employed WebHarvy for data mining, focusing on customer reviews of local restaurants. Their study aimed to determine seasonal variations' influence on restaurant success perceptions. The results indicated changing customer opinions about different facets of restaurant performance across different times of the year, highlighting a connection between season changes, product quality, and service standards.

Alatabi and Abbas [12] introduced a novel ML-based sentiment analysis system. They aimed to derive sentiment polarity from social media text. Their research proposed a system based on the Bayesian rough decision tree algorithm, demonstrating an accuracy exceeding 95% on social media data. Moreover, they emphasized that operations such as text preprocessing and feature selection significantly impact the system's accuracy, so more attention to this aspect could yield even more precise results.

Sert et al. [13] proposed a text analysis system that uses news and social media data to forecast stock market trends. They learned various prediction models in their study. They

used these well-trained models to predict the Dow Jones Index stock market trend and found that the proposed approach effectively predicted it. They got an accuracy of up to 70.90%. Nevertheless, we got a better accuracy on a different dataset.

Singh and Sarraf [14] proposed a system using a Random Forest classifier regarding the star rating of the reviewer. They presume that it does not always provide an accurate expression of sentiment. Furthermore, word reviews give an accurate representation of the item in question. This article focuses primarily on assessing customer reviews in the e-commerce industry. The system gives a Boolean review based on the comments instead of the scoring results, that is, whether or not the product is good, and users can analyze the product without reading all the reviews.

A combination of ML methods and NLP is used to analyze the sentiment of specific sentences introduced by Hossain et al. [15]. The accuracy of the CNN-LSTM combined architecture used on the dataset is 94.22%. The method employed CNN to grasp word representations, followed by LSTM to obtain more complex representations tailored for classification purposes.

Waghmare and Bhala [16] studied emotion classification, distinguishing between positive and negative sentiments. Their methodology involved explicit and implicit analysis of nouns and noun phrases to extract features from opinion texts. This process included a thorough review and elimination of irrelevant and redundant text. Furthermore, they proposed a deep neural network that demonstrated superior accuracy and efficiency in classification and extraction, outperforming various traditional ML algorithms. In contrast, the experiments conducted in the current study reveal a different outcome, where ML algorithms exhibited enhanced performance compared to DL methods.

A literature review indicates that Twitter has been a popular platform for conducting sentiment analysis, with a considerable focus on the Arabic language. However, there appears to be a research gap specifically addressing Twitter sentiment analysis concerning online learning. To the best of our knowledge, no previous work has explored the use of DL algorithms on Arabic tweets in the context of online learning. This gap presents a unique opportunity for current research to contribute novel insights and understandings in this area.

3. METHODOLOGY

The methodology employed in this study encompasses a series of steps designed to conduct Arabic Sentiment Analysis on the collected tweets, as depicted in Figure 1. These steps range from data collection to a comprehensive evaluation of the classifier models.

3.1 Data collection

The acquisition of tweets formed the foundation for creating a dataset amenable to sentiment analysis. The data utilized herein were the same as those in the study conducted by Alqaan and Qamar [5] during 2020-2021, coinciding with the adoption of the Madrasti platform for online education in Saudi Arabia. Initially, 100,000 Arabic tweets were gathered. Following a comprehensive filtration process, which included the exclusion of non-Arabic tweets, those containing advertisements, and tweets unrelated to online learning or the Madrasti platform, the dataset was reduced to 7,115 tweets.

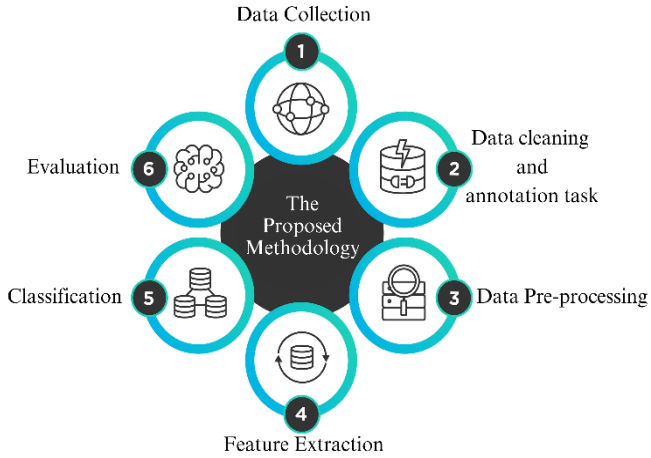

Figure 1. Major steps of the proposed methodology

3.2 Data cleaning and annotation task

3.2.1 Data cleaning

Given the typically noisy nature of Twitter data, a comprehensive cleaning process is usually required to ensure data accuracy and quality. Alqaan and Qamar [5] detailed a process that entailed eliminating punctuation and extra spaces and correcting spelling mistakes. After these cleaning operations, the dataset was refined down to 7,095 tweets.

3.2.2 Annotation

The cleaned data underwent sentiment annotation, which involved identifying emotional content in the text. This task was carried out manually by a team comprising three computer science graduates, all native Arabic speakers. The decision to involve three annotators, as opposed to two, facilitated the application of majority voting to resolve discrepancies in the annotation. These annotators were provided with detailed guidelines. The tweets were categorized into positive, negative, or neutral sentiments. The distribution of these categories is illustrated in Figure 2, with most tweets (3,917) classified as negative. The positive and neutral categories comprised 1,605 and 1,573 tweets, respectively.

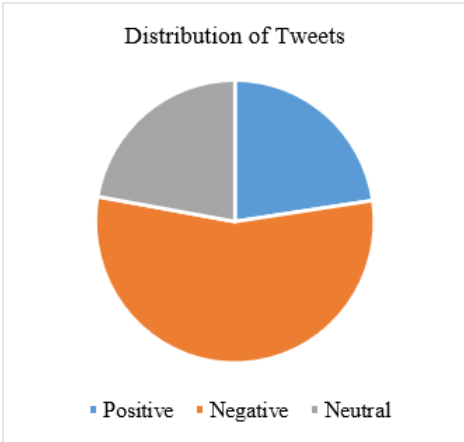

Figure 2. Distribution of tweets in three classes

3.3 Balancing the dataset

The dataset comprised three distinct sentiment classes as identified by the annotators. The distribution of tweets across

these categories, as illustrated in Table 1, indicates a predominant classification of tweets as Negative. This imbalance in class distribution leads to an imbalanced data issue, a common challenge affecting many datasets, necessitating strategies to mitigate the class disparity.

3.3.1 Oversampling

SMOTE is a widely used method for addressing class imbalance. This technique generates synthetic data by analyzing the feature spaces of existing minority instances. The effect of the SMOTE application on the dataset distribution is detailed in Table 1. Initially, most tweets were categorized as Negative, indicating a high level of imbalance. Implementing SMOTE resulted in a balanced dataset, augmenting the minority classes without reducing the number of instances in the majority class.

Table 1. Effect of applying the SMOTE technique on the data distribution

Class	Original Imbalanced Data	After Applying SMOTE
Positive	1605 tweets (22.6%)	3917 tweets (33.3%)
Negative	3917 tweets (55.2%)	3917 tweets (33.3%)
Neutral	1573 tweets (22.2%)	3917 tweets (33.3%)

3.3.2 Under-sampling

Random under-sampling was employed to address dataset imbalance further. This method randomly eliminates instances from the majority class, thereby balancing the dataset at each classification level. The data distribution before and after applying under-sampling is depicted in Table 2. Through this process, the dataset achieved a balanced state; however, it necessitated the removal of numerous instances from the majority class.

Table 2. Effect of applying the under-sampling technique on the data distribution

Class	Original Imbalanced Data	After Performing Under-Sampling
Positive	1605 tweets (22.6%)	1573 tweets (33.3%)
Negative	3917 tweets (55.2%)	1573 tweets (33.3%)
Neutral	1573 tweets (22.2%)	1573 tweets (33.3%)

3.4 Data preprocessing

Data preprocessing is a pivotal stage in any text classification system, enhancing data quality and improving the mining process's accuracy and efficiency. Common preprocessing steps in text analysis include text cleaning, tokenization, removal of stop words, stemming, and normalization [3].

3.4.1 Exclusion of punctuation marks

Punctuation symbols such as '$', '%', '&', and '_' are usually removed from text data, as they generally do not contribute inherent meaning. Their exclusion can lead to a reduction in data volume, enhancing the overall performance of the classification system [17].

3.4.2 Tokenization

Tokenization, also called segmentation, involves dividing words into their base morphemes, as exemplified in Table 3. Identifying and isolating these individual units is crucial for subsequent processing steps. In this process, whitespaces within a string are used as delimiters to separate words.

Table 3. Sample of tweets before and after the tokenization process

Tweet after Removing the Punctuation	Tweet after Tokenization
قيادة حكيمة حرصها الدائم على سلامة أبنائها الطلاب والطالبات من فيروس كورونا وتأتي توجيهات ولاة الأمر حفظهم الله بصدور الأمر السامي باستمرار مسيرة التعليم عن بعد عبر منصة مدرستي حفاظاً على صحة الجميع	قيادة, حكيمة, حرصها, الدائم, على, سلامة, أبنائها, الطلاب, والطالبات, من, فيروس, كورونا, وتأتي, توجيهات, ولاة, الأمر, حفظهم, الله, بصدور, الأمر, السامي, باستمرار, مسيرة, التعليم, عن, بعد, عبر, منصة, مدرستي, حفاظاً, على, صحة, الجميع]

3.4.3 Elimination of Arabic stopwords

Stop words, the most frequently occurring words in any language, are often redundant in text classification tasks. In Arabic, common stop words include (إذا, على, ذلك, غير) (If, on, that, not). However, some stop words such as (لا, لكن, لما, ليس) (عدا, ما) (No, but, why, not, except, what) may alter the sentence's meaning if removed. As per Alqaan and Qamar [5], the stop word list used in this study was modified to exclude common Arabic stop words while adding others more pertinent to this research.

3.4.4 Stemming

Stemming, a process of reducing words to their root form, is integral to text preprocessing. This step consolidates various word forms into a single representation, e.g., the words runs, running, and runner can be reduced to the root word run. This study utilized the Python ISRIStemmer library to stem Arabic words. Table 4 presents some examples of tweets after stemming, with inputs being tokens from which stop words have been removed.

Table 4. Sample of the applied stemming

Tweet after Removing the Arabic Stop Words	Tweet after Stemming
[قيادة, حكيمة, حرصها, الدائم, سلامة, أبنائها, الطلاب, والطالبات, فيروس, كورونا, وتأتي, توجيهات, ولاة, الأمر, حفظهم, الله, بصدور, الأمر, السامي, باستمرار, مسيرة, التعليم, عبر, منصة, مدرستي, حفاظاً, صحة, الجميع]	[قيد, حكم, حرص, دئم, سلم, بنئ, طلب, طلب, يرس, كورو, وتأ, وجه, ولة, امر, حفظ, الل, صدر, امر, سمي, باستمرار, سير, علم, عبر, نصة, درس, حفظ, صحة, جمع]

3.4.5 Normalization

Table 5. Effect of preprocessing on a sample tweet

Before Preprocessing	After Preprocessing
تحديث #منصة_مدرستي الجديد بنسبه للحضور والغياب سيء جدا الأول افضل.	حدث, نصة, درس, جدد, نسب, حضر, غيب, سيء, جدا, اول, فضل

The primary aim of normalization in text processing is to uniformly represent various forms of Arabic characters. The normalization in this research was executed per the methods outlined by Alqaan and Qamar [5], including substituting

certain Arabic characters (e.g., converting ؤ to و) and removing duplicate letters. The impact of this preprocessing step on a sample tweet, with stop words already removed, is illustrated in Table 5.

3.5 Classification algorithms

This study employs various ML and DL techniques for creating classification models. These include KNN, RF, SVM, and LSTM. These algorithms were selected based on their efficacy in recent related works and widespread use in classification tasks. Python serves as the programming language for implementing these classification tasks.

KNN is acknowledged as a fundamental yet essential algorithm in ML [18]. It classifies an instance based on the majority class of its K nearest neighbors [19]. The classification of a sample is determined by its proximity to other samples within a particular category [20].

SVM is recognized for its robustness in building classifiers. Its primary objective is to establish a decision boundary between two classes, which facilitates label predictions based on one or more feature vectors [21].

On the other hand, the RF Classifier is widely employed due to its versatility and simplicity. It creates a set of decision trees, and the number of trees, known as the "n estimator," influences the final prediction through a majority vote. Increasing the number of trees can enhance prediction accuracy and stability, although it may also lead to longer computation times [22].

The number of neighbors (K) for KNN was chosen as 4, whereas the SVM kernel is linear. Similarly, 150 trees are employed for the RF classifier.

LSTM is widely used in classification tasks and includes specialized units to address the vanishing gradient issue. In LSTM architecture, gating mechanisms regulate the information flow within the network, and a memory cell is employed to store data for extended durations. The gating mechanisms manage the data flow into and out of the cell, enabling the network to selectively retain or discard significant or insignificant information to the current task. The memory cell stores data through the utilization of three gates: the 'Forget Gate,' which decides what previous state cell information should be retained and what should be discarded; the 'Input Gate,' which determines the information that should be entered into the cell state, and the 'Output Gate,' which controls the outputs. Due to its ability to overcome recurrent network training issues, the LSTM network is regarded as one of the most successful RNN architectures [23].

In this study, the LSTM model comprises an embedding layer, succeeded by an LSTM layer housing 128 neurons. Subsequently, a fully connected layer equipped with a sigmoid activation function calculates the output. The ADAM optimizer adjusts the learning rate, and the binary cross-entropy function is the objective function. The model underwent training with 10 epochs, incorporating early stopping to prevent overfitting. Hyperparameter tuning was conducted by varying the number of epochs, and the model was evaluated using 20% of the dataset.

4. EXPERIMENTS AND RESULTS

The evaluation of NLP tasks often relies on several key metrics, including accuracy, precision, recall, and F-measure. These metrics are crucial in assessing the performance of classification models. The results are represented in a confusion matrix, as Table 6 [24] illustrates.

Table 6. Confusion table

	Actual	
Prediction	Positive class	Negative class
Positive class	True Positives	False Positives
Negative class	False Negatives	True Negatives

In this study, a 10-fold Cross-validation approach was utilized. Accuracy, a fundamental metric, is computed as per the formula provided in Eq. (1) [25]:

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \quad (1)$$

Precision, which assesses the model's ability to correctly identify positive instances among all instances classified as positive, is calculated using the formula in Eq. (2) [25]:

$$Precision = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \quad (2)$$

Recall, also known as sensitivity, measures the model's ability to identify all positive instances. This is quantified as shown in Eq. (3):

$$Recall = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (3)$$

The F-measure, or F1-score, integrates precision and recall into a singular metric, offering a balanced view of both. It is calculated as the harmonic mean of precision and recall, as demonstrated in Eq. (4):

$$F1-score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

4.1 Evaluation of the multi-classification after applying SMOTE

The model's performance was evaluated using the same metrics as those applied to the traditional model on an unbalanced dataset, as reported by Alqaan and Qamar [5]. Notably, the earlier work did not present macro-average results. Table 7 displays the classification outcomes using different ML algorithms after applying SMOTE, with the ± sign denoting standard deviation.

Post-application of SMOTE to the data, there was a notable improvement in all evaluation measures, particularly significant in most cases, compared to the unbalanced dataset results by Alqaan and Qamar [5]. This enhancement was especially evident in the performance of the minority classes. For instance, using the SVM, the results ranged from 0.81 to 0.91 across all measures, except for a decrease in recall for the negative class to 0.79. However, a limitation of SMOTE is its potential to oversample uninformative or noisy samples.

In the case of the KNN algorithm, recall and F1-score for the negative class demonstrated lower results than the original data before applying SMOTE [5]. A similar trend was observed in the recall for the negative class using RF.

The best classification method was selected based on the model's ability to predict all classes accurately. RF, when applied with SMOTE, exhibited the most favorable results, achieving 0.89, 0.86, and 0.86 in Macro-Avg (Precision), Macro-Avg (Recall), and Macro-Avg (F1-score), respectively. It consistently performed better across all sentiment classes, with F1-scores ranging between 0.83 and 0.90. This could be attributed to the general effectiveness of SMOTE with tree-based classifiers [26].

Overall, RF emerged as the most robust model, delivering consistently superior performance across different sentiment classes. The effectiveness of SMOTE with tree-based algorithms, like RF, is a likely contributing factor. While SVM also showed commendable performance, it may require further tuning for the negative class. KNN, despite its strengths in certain aspects, faced challenges in recall for negative sentiment classification, affecting its overall performance. These findings suggest that future research could benefit from exploring tree-based methods such as ID3 and C4.5.

4.2 Evaluation of the multi-classification after applying the random under-sampling

Table 8 shows the results for different metrics of the multi-classification after applying the under-sampling technique. One can observe notable improvements in the results for the neutral class as compared to the unbalanced data, as documented by Alqaan and Qamar [5]. For instance, in the case of SVM, the recall for the neutral class increased from 0.06 to 0.48, and the F1-score improved from 0.11 to 0.52. However, it is important to note that the overall performance of SVM declined compared to the results obtained with SMOTE. Similarly, in the RF classifier, the recall for the neutral class escalated from 0.03 to 0.65, and the F1-score rose from 0.06 to 0.75.

Conversely, the results for other classes exhibited some deterioration. This outcome can be attributed to the nature of random under-sampling, which entails the removal of random samples from the majority class in the training dataset to achieve a more balanced distribution. This process can lead to the loss of certain specific instances, potentially adversely impacting the model's performance. For example, the precision of the positive class using the SVM classifier decreased from 0.77 to 0.69, and the recall of the negative class in the RF classifier dropped from 0.97 to 0.87.

In summary, RF maintains its position as the most robust model, exhibiting consistently superior performance across all classes. Meanwhile, KNN continues to show the lowest overall effectiveness, particularly struggling to classify neutral sentiments.

4.3 Impact of balancing techniques

Table 9 provides a comprehensive overview of the effects of data balancing on model performance. In this comparison, the focus is exclusively on the F1-scores for a fair assessment. The unbalanced results are taken directly from the study by Alqaan and Qamar [5].

Table 7. Evaluation results of the multi-classification after applying SMOTE

Model	Class	Precision	Recall	F1-score	Macro-Avg (Precision)	Macro-Avg (Recall)	Macro-Avg (F1-score)
SVM	Positive	0.89±.10	0.91±0.18	**0.90±0.11**	0.86±0.13	0.85±0.14	0.85±0.14
	Negative	0.86±0.23	0.79±0.23	0.81±0.14			
	Neutral	0.82±0.13	0.84±0.27	0.83±0.19			
RF	Positive	**0.91±0.12**	0.90±0.24	**0.90±0.13**	**0.89±0.15**	**0.86±0.20**	**0.86±0.19**
	Negative	0.84±0.37	**0.86±0.24**	**0.83±0.20**			
	Neutral	**0.91±0.11**	0.81±0.43	**0.84±0.28**			
KNN	Positive	0.72±0.06	**0.95±0.10**	0.82±0.07	0.77±0.05	0.65±0.07	0.55±0.06
	Negative	**0.99±0.04**	0.06±0.04	0.11±0.06			
	Neutral	0.59±0.09	**0.95±0.10**	0.73±0.09			

Table 8. Evaluation results of the multi-classification after applying the under-sampling technique

Model	Class	Precision	Recall	F1-score	Macro-Avg (Precision)	Macro-Avg (Recall)	Macro-Avg (F1-score)
SVM	Positive	0.69±0.06	0.74±0.06	0.72±0.05	0.64±0.05	0.65±0.04	0.64±0.05
	Negative	0.66±0.05	0.72±0.04	0.69±0.04			
	Neutral	0.57±0.08	0.48±0.09	0.52±0.08			
RF	Positive	**0.84±0.04**	**0.86±0.03**	**0.85±0.03**	**0.81±0.03**	**0.79±0.03**	**0.79±0.03**
	Negative	**0.71±0.04**	**0.87±0.03**	**0.78±0.03**			
	Neutral	**0.88±0.05**	**0.65±0.06**	**0.75±0.04**			
KNN	Positive	0.54±0.03	0.74±0.05	0.62±0.03	0.55±0.04	0.55±0.04	0.54±0.04
	Negative	0.61±0.06	0.60±0.07	0.60±0.06			
	Neutral	0.49±0.06	0.32±0.04	0.39±0.04			

Table 9. Impact of balancing the dataset on the F1-score

Model	Class	Unbalanced	Under-Sampling	SMOTE
SVM	Pos.	0.61 ± 0.16	0.72±0.05	**0.90±0.11**
	Neg.	0.79 ± 0.03	0.69±0.04	**0.81±0.14**
	Neut.	0.11 ± 0.11	0.52±0.08	**0.83±0.19**
RF	Pos.	0.61 ± 0.18	0.85±0.03	**0.90±0.13**
	Neg.	0.78 ± 0.03	0.78±0.03	**0.83±0.20**
	Neut.	0.06 ± 0.06	0.75±0.04	**0.84±0.28**
KNN	Pos.	0.52 ± 0.12	0.62±0.03	**0.82±0.07**
	Neg.	**0.71 ± 0.09**	0.60±0.06	0.11±0.06
	Neut.	0.27 ± 0.07	0.39±0.04	**0.73±0.09**

The data indicate that applying SMOTE generally leads to significant enhancements in the results across all ML techniques, typically surpassing the outcomes achieved with random under-sampling. This improvement can be attributed to the mechanism of SMOTE, which, instead of removing instances, duplicates some instances in minority classes. Conversely, random under-sampling involves the elimination of instances from the majority class, which may inadvertently result in the loss of critical instances.

In the specific case of KNN, it was observed that SMOTE did not effectively improve results for the negative class. Similarly, the random under-sampling approach also failed to enhance outcomes for KNN. This limited impact of SMOTE on the KNN classifier might be due to the general compatibility of SMOTE with tree-based classifiers like RF rather than KNN.

4.4 LSTM

In this study, the LSTM network was fine-tuned with various hyperparameter settings to optimize its performance for the given NLP task. A word embedding layer, a common feature in NLP models, was employed to generate word embeddings, incorporating a vocabulary size of 26,870. The specific hyperparameters for the different LSTM models are detailed in Table 10.

Table 10. Values of the hyperparameters for different LSTM models

Parameters	Values
Activation function	Sigmoid
Batch size	32 – 128
Dropout rate	0.2
Epochs	10 – 60
Loss function	Binary cross-entropy
LSTM layers	1
Nodes	100
Optimizer	ADAM
Recurrent dropout	0.2

Figure 3 presents the accuracy trends for training and validation sets during the LSTM training process. The highest accuracy achieved on the validation set was 0.556, recorded in the 7th epoch, while the training set reached its peak accuracy of 0.591 in the 10th epoch. Initially, an increase in epochs correlated with rising accuracies. However, after only 3 epochs, a decline in validation accuracy was noted, indicating potential overfitting. In contrast, the training accuracy started to decrease after 6 epochs. The accuracy discrepancy between the training and validation sets was initially over 15%, but this gap narrowed to just over 5% in the later epochs.

In a subsequent experiment with increased epochs to 20 while keeping other hyperparameters unchanged, a marked increase in the experiment duration was observed. Figure 4 displays the accuracy curves for the training and validation sets under this extended epoch condition. Despite the increased number of epochs, the maximum accuracies for both training and validation sets remained consistent with those observed in the previous experiment (Figure 3). Notably, the gap between training and validation accuracies was smaller at the 20th iteration than in earlier epochs.

Further increasing the number of epochs to 30 did not improve results. Additionally, the accuracy of test data was recorded at only 0.528. These findings suggest that the LSTM model did not perform optimally on this dataset, with its results being less favorable compared to those achieved with traditional ML techniques.

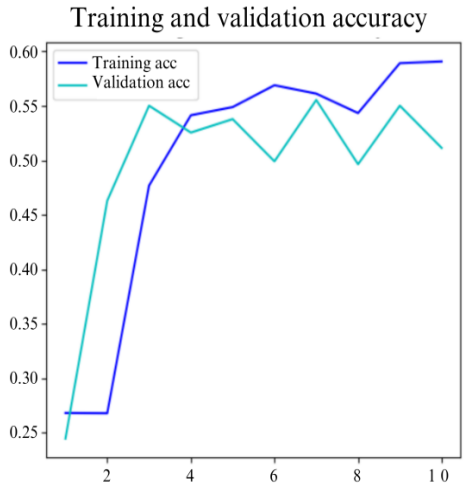

Figure 3. Training and validation accuracies with a batch size of 128 and maximum iterations of 10

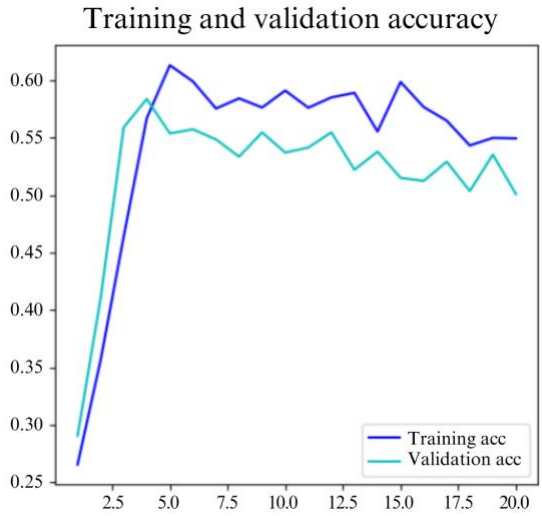

Figure 4. Training and validation accuracies with a batch size of 128 and maximum iterations of 20

5. CONCLUSIONS

This study has established a balanced dataset of Arabic tweets for sentiment analysis, focusing on online learning. The dataset, enriched with individuals' reviews and opinions, was analyzed using ML techniques (SVM, KNN, RF) and a DL approach (LSTM). The aim was to identify patterns in the data that could enhance the quality of online learning experiences. The performance of these algorithms was assessed using precision, recall, and F-measure.

The initial challenge of data imbalance was addressed using the SMOTE and random under-sampling. Macro-average measures ranged from 54% to 89%. The results indicated that SMOTE generally yielded better outcomes across most classes, particularly with the RF classifier, which surpassed both SVM and KNN. The highest F1-score was recorded with RF, achieving 86% using SMOTE and 79% with under-sampling. SMOTE's effectiveness can be attributed to its preservation of

information, a contrast to the loss inherent in under-sampling. Moreover, RF's superior performance is likely due to the general compatibility of SMOTE with tree-based classifiers. While the dataset was developed during the COVID-19 pandemic, the insights apply to online learning in a broader context. Traditional ML algorithms demonstrated greater efficacy than LSTM. Augmenting LSTM layers and experimenting with various optimizers might enhance LSTM's performance.

Future research directions include expanding the dataset to improve prediction accuracy and reliability. A larger dataset could mitigate the risk of losing critical information in under-sampling. Exploration of diverse ML algorithms, particularly tree-based classifiers, in conjunction with SMOTE, will be prioritized due to their promising results. Additionally, other DL methods, such as convolutional neural networks and recurrent neural networks, will be investigated to broaden the scope of the analysis.

ACKNOWLEDGMENT

The authors gratefully acknowledge Qassim University, represented by the Deanship of Scientific Research, on the financial support for this research under the number (COC-2022-1-2-J-29921) during the academic year 1444 AH / 2022.

REFERENCES

[1] Ferdianto, T., Desak, G.G.F.P., Lena. (2018). A comparative study of teaching styles in the online learning environment. In International Conference on Information Management and Technology, Special Region of Yogyakarta, Indonesia, pp. 25-30. https://doi.org/10.1109/ICIMTech.2017.8273505

[2] Ramanathan, V., Meyyappan, T. (2019). Twitter text mining for sentiment analysis on people's feedback about Oman tourism. In 4th MEC International Conference on Big Data and Smart City, Muscat, Oman, pp. 1-5. https://doi.org/10.1109/ICBDSC.2019.8645596

[3] Al-Harbi, W.A., Emam, A. (2015). Effect of Saudi dialect preprocessing on Arabic sentiment analysis. International Journal of Advanced Computer Technology, 4(6): 91-99.

[4] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. Nature, 529: 484-489. https://doi.org/10.1038/nature16961

[5] Alqaan, S.E., Qamar, A.M. (2022). Utilizing sentiment analysis to enhance the quality of online learning. In Fifth National Conference of Saudi Computers Colleges, Jeddah, Saudi Arabia, pp. 41-46. https://doi.org/10.1109/NCCC57165.2022.10067560

[6] Dhawan, S., Singh, K., Chauhan, P. (2019). Sentiment analysis of Twitter data in online social network. In 5th International Conference on Signal Processing, Computing and Control, Solan, India, pp. 255-259. https://doi.org/10.1109/ISPCC48220.2019.8988450

[7] Wongkar, M., Angdresey, A. (2019). Sentiment analysis using Naïve Bayes algorithm of the data crawler: Twitter. In Fourth International Conference on Informatics and Computing, Semarang, Indonesia, pp. 1-5. https://doi.org/10.1109/ICIC47613.2019.8985884

[8] Ullah, M.A., Marium, S.M., Begum, S.A., Dipa, N.S. (2020). An algorithm and method for sentiment analysis using the text and emoticon. Information & Communications Technology Express, 6(4): 357-360. https://doi.org/10.1016/j.icte.2020.07.003

[9] Almalki, J. (2022). A machine learning-based approach for sentiment analysis on distance learning from Arabic Tweets. PeerJ Computer Science, 8: e1047. https://doi.org/10.7717%2Fpeerj-cs.1047

[10] Aljabri, M., Chrouf, S.M.B., Alzahrani, N.A., Alghamdi, L., Alfehaid, R., Alqarawi, R., Alhuthayfi, J., Alduhailan, N. (2021) Sentiment analysis of Arabic tweets regarding distance learning in Saudi Arabia during the COVID-19 pandemic. Sensors, 21(16): 5431. https://doi.org/10.3390/s21165431

[11] Yu, C.E., Zhang, X. (2020). The embedded feelings in local gastronomy: A sentiment analysis of online reviews. Journal of Hospitality and Tourism Technology, 11(3): 461-478. https://doi.org/10.1108/JHTT-02-2019-0028

[12] Alatabi, H.A., Abbas, A.R. (2020). Sentiment analysis in social media using machine learning techniques. Iraqi Journal of Science, 61(1): 193-201. https://doi.org/10.24996/ijs.2020.61.1.22

[13] Sert, O.C., Şahin, S.D., Özyer, T., Alhajj, R. (2020). Analysis and prediction in sparse and high dimensional text data: The case of Dow Jones stock market. Physica A: Statistical Mechanics and its Applications, 545: 123752. https://doi.org/10.1016/j.physa.2019.123752

[14] Singh, S.N., Sarraf, T. (2020). Sentiment analysis of a product based on user reviews using random forests algorithm. In 10th International Conference on Cloud Computing, Data Science & Engineering, Noida, India, pp. 112-116. https://doi.org/10.1109/Confluence47617.2020.9058128

[15] Hossain, N., Bhuiyan, M.R., Tumpa, Z.N., Hossain, S.A. (2020). Sentiment analysis of restaurant reviews using combined CNN-LSTM. In 11th International Conference on Computing, Communication and Networking Technologies, Kharagpur, India, pp. 1-5. https://doi.org/10.1109/ICCCNT49239.2020.9225328

[16] Waghmare, K.A., Bhala, S.K. (2020). Survey paper on sentiment analysis for tourist reviews. In International Conference on Computer Communication and Informatics, Coimbatore, India, pp. 22-25. https://doi.org/10.1109/ICCCI48352.2020.9104197

[17] Kulkarni, A., Shivananda, A. (2019). Natural Language Processing Recipes. Springer [Online]. Available: https://link.springer.com/book/10.1007/978-1-4842-4267-4.

[18] Okfalisa, Gazalba, I., Mustakim, Reza, N.G.I. (2018). Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification. In 2nd International conferences on Information Technology, Information Systems and Electrical Engineering, Yogyakarta, Indonesia, pp. 294-298. https://doi.org/10.1109/ICITISEE.2017.8285514

[19] Agrawal, R. (2014). K-nearest neighbor for uncertain data. International Journal of Computer Applications, 105(11): 13-16. https://doi.org/10.5120/18420-9714

[20] Wang, J., Neskovic, P., Cooper, L.N. (2006).

Neighborhood size selection in the k-nearest-neighbor rule using statistical confidence. Pattern Recognition, 39(3): 417-423. https://doi.org/10.1016/j.patcog.2005.08.009
[21] Noble, W.S. (2006). What is a support vector machine? Nature Biotechnology, 24(12): 1565-1567. https://doi.org/10.1038/nbt1206-1565
[22] Müller, A.C., Guido, S. (2016). Introduction to Machine Learning with Python. O'Reilly Media, Inc.
[23] Sarker, I.H. (2021). Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. SN Computer Science, 2: 420. https://doi.org/10.1007/s42979-021-00815-1
[24] Han, J., Kamber, M., Pei, J. (2012). Data Mining: Concepts and Techniques. Elsevier Inc.
[25] Olson, D.L., Delen, D. (2008). Advanced data mining techniques. Springer Science & Business Media. https://doi.org/10.1007/978-3-540-76917-0
[26] Ishaq, A., Sadiq, S., Umer, M., Ullah, S., Mirjalili, S., Rupapara, V., Nappi, M. (2021). Improving the prediction of heart failure patients' survival using SMOTE and effective data mining techniques. IEEE Access, 9: 39707-39716. https://doi.org/10.1109/ACCESS.2021.3064084