Sentiment Analysis of Twitter Data on Smart City Services using BERT Model

Abrar Albeladi†, and Turki Alghamdi†

†Department of Computer Science and Engineering, College of Computers and Information Systems,
Umm Al-Qura University, 46477, Saudi Arabia;

Summary
Sentiment analysis of online platforms has emerged as a significant research area in various domains, including smart cities. Governments in many countries aim to improve the quality of service provisions to ensure sustainability and create smart city systems. The use of technologies to support this goal makes cities a great source of data. Social media platforms, particularly Twitter, offer a rich source of data, as users share their opinions, emotions, and evaluations about events and services. Governments can utilize these citizens' opinions to enhance services provisions. However, the abundance of noisy data on social media platforms poses a challenge in identifying relevant information, hindering timely responses to citizen feedback. This study focuses on analyzing Arabic tweets related to smart cities from two Saudi cities to assist decision-makers in future planning. A three-class dataset with each class containing 1109, 771, and 8804 samples was used for experimentation. Two sets of experiments were conducted: one without Synthetic Minority Oversampling Technique (SMOTE) and the other with SMOTE. The experimental results without SMOTE revealed that linear support vector machine (SVM) and multinomial Naive Bayes (NB) achieved better performance. However, more promising results were obtained with SMOTE analysis, where Logistic Regression, Random Forest, Linear SVC, Gradient Boosting, Decision Tree, Voting, and Stacking algorithms demonstrated superior performance.

Keywords:
Sentiment analysis, Random forest algorithm, Smart cities, Arabic tweet datasets, SMOTE

1. Introduction

Recently, as surveys depicted, most people around the world are living in urban cities that demands swift and quality services. According to the United Nations Urbanization report [1], by 2050, about 64% of world's population will live in cities, therefore the urban population is increasing alarmingly. People move from rural areas to cities looking for better opportunities, jobs, education, health care, and better livelihood in general [2]. Governments recognize the rapid growth of cities population growth and the importance of meeting people's demands by making these cities more livable and sustainable. To ensure sustainability, governments tend implement innovative solutions for city services and utilities in all dimensions such as social, environmental, and economy. Hence, the concept of smart cities is on the rise alarmingly.

Several research works were conducted in conceptualizing and defining the smart city concept by its characteristics and tools [3]. The International Telecommunication Union (ITU) defines smart cities as innovative cities that employ Information and Communication Technologies (ICT) and other means to improve the quality of life, efficiency of urban operations and services, and competitiveness [4]. International Organization for Standardization (ISO) evaluates the smart city's performance from the perspectives of its inhabitants, administrators, and the surrounding environment. Taewoo and Theresa [5] identified three main dimensions of smart cities: smart governance and policies, educated people, and smart technologies. Moreover, smart cities include smart (economy, mobility, environment, health care, living, education, safety) systems.

ICT play a crucial role in achieving and facilitating smart city goals. The use of ICT, besides the huge population, makes smart cities produce enormous relevant data that comes from sensors in physical entities or from applications in mobile devices [6]. Governments and decision-makers need to collect and analyze this data in order to make useful insights into citizens' level of life and to evaluate cities' services. Based on the results of the analysis stakeholders will be able to assess and improve future services to make cities smarter and enhance the overall quality of life for residents.

The evolution of network services in smart cities makes social media platforms popular and widely used. Social media platforms have become a popular medium for people to express their thoughts, opinions, and share their daily activities with others. Social media data are

Manuscript received February 5, 2025
Manuscript revised February 20, 2025
https://doi.org/10.22937/IJCSNS.2025.25.2.7

unstructured, dynamic, vast, noisy, distributed, and challenging to collect and do analysis [7]. Social media mining assist governments in getting people's opinions and thoughts without the need for traditional surveys.

Opinion mining (sentiment analysis) is a process of studying people's opinions, feelings, evaluations, sentiments, and emotions toward services, events, products, organizations, and topics [8]. Millions of peoples are sharing their opinions about products or services online on daily basis to influence community by someone's opinion and thoughts. Hence, service providers go for leveraging social media comments and reviews to assess and improve their services. Since 2000s, sentiment analysis became an active research arena in Natural Language Processing (NLP) and is impacting several industries.

There are several challenges associated with sentiment analysis including computing costs, informal language, and differences across languages. Sentiment analysis depends on phrases and words that clearly express positive or negative sentiment using sentiment lexicons. However, relying on lexicons only gives less relievable sentiments. Furthermore, lexicon words can be used without a real sentiment as in question form statements [8].

The Arabic language is written from right to left without the use of capital letters for nouns, which is a needful for text mining. There are 28 letters in the Arabic alphabet. The Arabic language has three forms: Classical Arabic (CA) such as the Quran language, Modern Standard Arabic (MSA) such as the daily spoken Arabic in news and formal meetings, and dialectical Arabic which differs between Arabic regions and countries such as Egyptian, Gulf, and Iraqi [9]. Thus, Arabic language sentiment analysis must take into account the various dialects and the different types of free writing that make this field of study more challenging. The majority of Arabic conversations on social media are in regional dialects.

Twitter is indeed one of the most popular social media applications worldwide, boasting a substantial user base and a high volume of daily interactions. Over 500 million tweets are posted on Twitter every day, making it a platform for real-time information sharing and discussions [10,11]. Sentiment analysis of Twitter posts can be utilized to understand citizens' issues related to sustainability in order to plan how to solve them. Many previous researches have been done on different smart city dimensions and issues using Twitter data. Thus, we can consider Twitter a worthy data source and conduct data-driven studies such as sentiment analysis.

Before the discovery of oil in 1938 in Saudi Arabia, the Saudi economy was mostly supported by pilgrims making the Hajj and Umrah. During those times, the economy of the Kingdom of Saudi Arabia (KSA) was booming eventually, everyday living standard was less complicated, and economic growth was slow. The oil discovery takes the economy in a new direction. Since then, Saudi Arabia's economy has flourished with consistent development throughout the past decade. In 1960, Saudi Arabia's urban population rate was 31%, which grew to 79% in 2000. In 2022, 84.7% of Saudi Arabia's population lived in cities [12]. This population growth increases citizens' requirements to live a quality life. The urban transformation arise many challenges in housing, employment, sustainability, economy, transportation, and other fields. Vision 2030, as per many governments' belief, the world economy basis in cities that is on its way to transforming from an oil-based economy to a knowledge-based economy [13]. Based on the Institute for Management Development (IMD) issued report, the smart city index for 2021, Riyadh and Medina were ranked 23 and 73 respectively out of 118 cities [14].

Social media is used in smart cities to improve communication and engagement between the city and its citizens, to gather information and feedback, and to promote and market the city's initiatives and events. Saudi Arabia takes the fourth position on the number of Twitter users with 15.5 million users [15]. Users use Twitter to share their opinions, struggles, and stories in near real-time. Analyzing Saudi Twitter posts is a big opportunity to evaluate and improve the vision goals and initiatives of creating a more livable and sustainable country. Based on these facts, we aim to analyze Twitter posts in Saudi Arabia in Riyadh and Makkah in some smart city dimensions regarding to environment, people, living, and governance. The results will help in evaluating the current situation of Makkah.

Rapid urbanization in smart cities creates new issues that governments should handle. Governments need to evaluate and assess the provision of smart city services to ensure whether the sustainability goals are met. Some of the sustainability goals target citizens' satisfaction with the environment, education, citizen engagement, and tourists' opinions. The fastest and most reliable way is to directly communicate with citizens to address the issues. Analyzing Twitter will save valuable time and help decision-makers in viewing the current situation and improving it. Twitter users vary from official accounts, influencers, and rumor publishers. Thus, the reliability of Twitter as a data source needs to be addressed and studied. The lack of available Arabic datasets [9] encourages to build and publish a specialty dataset.

The main objective of the study is to construct a model for Arabic Twitter sentiment analysis. The address this general objective, the following specific objectives were achieved.

1. To construct and publish an Arabic corpus related to smart city extracted from Twitter in order to enrich resources and encourage future researchers in the area of smart city.
2. To construct a framework for Arabic Twitter sentiment analysis related to smart city.
3. Develop a robust deep-learning model to classify smart city datasets.

The present paper is arranged in the following manner: Section 2 explores the existing literature and background information related to the primary focus of the research, including the use of Twitter for sentiment analysis, and corpus production. Section 3 presents the proposed BERT model for the classification of Arabic tweet datasets for sentiment analysis. Section 4 describes the performance evaluation metrics used during the process of evaluating the machine learning models. Section 5 provides the main results and the discussion. Finally, Section 6 summaries the main conclusions and future work.

2. Related Work

This section explores the existing literature related to the primary focus of the research, including the use of Twitter for sentiment analysis, and corpus production. The aim is to gain a comprehensive understanding of these key concepts. Several researchers have studied smart city dimensions and characteristics using Twitter sentiment analysis. The sentiment analysis is based on one of three main approaches: a machine-learning approach, a lexicon-based approach, and a hybrid approach [16]. The lexicon-based approach uses collections of words that are marked with their polarity and the degree of sentiment they convey to identify the polarity of new data. While the machine-learning approach uses trained models to predict the polarity of test data. The hybrid approach combines the previous two approaches.

Wella and Christian [17] exploit Twitter as a data source to analyze tourists' sentiments in Bali by using four supervised machine learning algorithms and comparing their results. They used Decision Tree (DT), SVM, K-Nearest Neighbor (KNN), and NB to identify the best algorithm in sentiment analysis. They used Twitter API and R Studio to extract and collect tweets containing the word "Bali" over two months. The data was pre-processed by removing usernames, hashtags, emojis, punctuation, Uniform Resource Locator (URLs), and repeated words. Then, specialists labeled the tweets as positive or negative manually. Researchers used the Rapidminer platform for the classification process, furthermore, they used some packages and methods to clear and prepare the data. For feature extraction, researchers used the Term Frequency-Inverse document frequency method to identify the importance of a word. Then, the four algorithms were applied for classification. SVM has the highest Area Under Curve (AUC) value while K-NN has the highest accuracy value. Finally, the popular words and sentiments were visualized using R studio.

Abdullah et. al [18] exploits Twitter data to figure out people's opinions about companies and measure company's reputation. This data is a tremendous asset for businesses and policy-makers. They studied telecommunications companies in Saudi Arabia as a case study. Since Twitter users of different Telecom providers tend to have different perspectives, it is necessary to find a new method of computing the polarity of Arabic sentiment and reach in reliable reputation score for each service provider. A hybrid approach of verbal and non-verbal opinions was to better gauge public sentiment. Non-verbal opinions include a favorite tweets and retweets. Data were collected from the official accounts of the telecom companies using Twitter API. Then, the data was pre-processed and normalized to remove irrelevant contents. The Mazajak tool was used to label the data as positive or negative. However, using tools to automatedly annotate the data may not be accurate. The analysis went through two phases: (a) Using machine learning models on the verbal data and (b) Calculating the polarity equation using the number of retweets and likes (i.e. non-verbal data). The output of the second phase was used in reputation calculation. The results of the reputation were enhanced with the use of the polarity scoring equation. Results show that SVM outperforms other classifiers while Decision Tree performs the worst.

Alsulami and Mehmood [19] study users' opinions about government decisions in Saudi Arabia. The study relied on analyzing the emotions expressed in tweets from Saudis regarding a decision made by the Ministry of Education. It aimed to find out users' acceptance and expectations to help the ministry in perceiving citizens' feedback. Researchers suggested a seven-phase methodology. The first phase was to identify the related keywords. Secondly, the words were used to run a query on Twitter API and the result was stored in the SAP HANA platform. Thirdly, to create two Arabic dictionaries: one to detect the level of acceptance and the other to detect expectations. The first dictionary has three categories: negative, neutral, and positive while the second dictionary has three. categories: expectations related to faculty, expectations related to students, and expectations related to administration. Fourthly, preparing configuration files. Fifth, to create full-text indexes of the posts. Sixth, applying

calculation view with filter expressions using SAP HANA. Seventh, visualizing the results using SAP Lumira.

Hassonah [20] proposed an effective hybrid model for sentiment analysis of diverse Twitter topics by combining filter and evolutionary wrapper techniques. Researchers used REST API with specific words and hashtags to retrieve data in different domains such as movies, products rating, restaurant reviews, and public opinions, and store them in nine data sets. After tweets' extraction, they data processing carried out such as deletion of any irrelevant information such as number of favorites (likes) and retweets, and part of the tweeter's profile data. The TF-IDF was used as a weighting strategy to come up with feature set. Hybrid machine learning approach that combines the Multi-Verse Optimizer (MVO) and ReliefF algorithms with the SVM classifier were used for future selection and dimensionality reduction.

Annett et. al. [21] compares the performance of lexicon-based and machine learning models. The findings show that accuracy exceeds in lexicon-based methods. However, machine learning methods perform better with a large annotated training dataset. Table 1 shows the Non-Arabic Twitter sentiment analysis.

Table 1: Twitter Sentiment Analysis

Cite	SC Domain	Techniques	Results
Salas et. al. [22]	Traffic/ Smart Transportation	SentiStrength tool	Measuring the sentiment intensity helps in estimating the real situation.
Li et. al. [23]	Citizens' mood/ Smart People	Emoji ranking system + Multinomial Naive Bayes	Accuracy enhanced with considering emojis.
Steven et. al. [17]	Tourism/ Smart Environment	ML algorithms: SVM, DT, KNN, NB algorithms	SVM has the highest AUC value while K-NN has the highest accuracy value.
Aysha et. al. [24]	Traffic/ Smart Environment	TextBolb python library	No results mentioned.
Musto et. al. [25]	Earthquick impact on citizens/Smart Environment	lexicon-based analyzer (SenticNet) + ML algorithms	Effectiveness of a multi- disciplinary methodology Using semantic and sentiment analysis
Ali et. al. [26]	Political / Smart Governance	Sentiment analyzers: TextBlob, Senti-WordNet, and WSD + ML algorithms: SVM and NB to test the analyzers	WSD has the highest accuracy with NB while it gives the same accuracy as TextBlob with SVM.
Manguri et. al. [27]	Healthcare / Smart People	TextBlob + Emotional Guidance Scale	The use of the scale helps in reflecting people's emotional state during pandemics.
hassonah et. al. [20]	General topics	Features extraction: MVO and ReliefF+ ML algorithms: SVM	Features reduction contributed in enhancing sentiment analysis accuracy.
Arambepola et. al. [28]	Education / Smart People	TF-IDF + TextBlob	Sentiment analysis can aid in statistical analysis.
Alsulami et. al. [19]	Education / Smart Governance	SAP HANA platform	No results mentioned
Alayba et. al. [29]	Healthcare / Smart people	ML algorithms: NB,SVM, and LR +Deep learning algorithms: DNN and CNNs algorithms	SVM and CNN gives higher accuracy than other algorithms
Abdullah et. al. [18]	Telecommunication / Smart Governance	ML algorithms: SVM, DT, and NB	SVM has best results while DT has worst results
Alomari et. al. [30]	General topics	ML algorithms: SVM and NB with different term weighting techniques, stemming techniques, and N-gram methods	Best classification was obtained by using SVM classifier with stemming, TF-IDF weighting scheme, and Bi-grams
Aljameel et. al. [31]	Healthcare / Smart people	Bigram and un-igram with and without the TF-IDF technique + ML algorithms: SVM, KNN, and NB	SVM gave highest accuracy with Bi-gram and TF-IDF
Hashedi et. al. [32]	COVID-19 conspiracy / Smart people	Two word embedding models + with and without SMOTENC method + single-based and ensemble-based	ML algorithms word embedding, SMO-TENC, and ensemble-based algorithms

"

The researchers examined the features to determine for their subjective, objective, or emoticon for sentiment analysis. The filter approach uses the ReliefF for feature selection that arranges features according to their level of significance. On the other hand, the wrapper approaches use evolutionary algorithms to search a subset of features and find the best parameters for the SVM classifier, even though it is costly in terms of computations. According to the empirical evidence, their model outperforms them although 96% fewer features are used overall. They also highlighted hybrid models' capabilities and came to the conclusion that hybrid models might outperform all other models with a careful selection of hyperparameters and adequate architecture.

3. The Proposed Methodology

This section presents the developed framework for the classification of Arabic tweet datasets for sentiment analysis using machine learning models. The proposed method considers Arabic tweets on smart cities applicable for the smart city detection task. In a text processing task, there are several interlinked and complementary preprocessing steps including lexical analysis, elimination of punctuation marks, stop words removal, tokenization, and stemming to root words [33]. Pre-processing steps were applied to the data to convert words into numerical features that suit the specific machine-learning algorithm under consideration. It is evident that pre-processing steps in NLP depend on the domain in general and on the problem in particular. In this study, an Arabic dataset from Twitter platform with three labels is employed to develop, validate and test the proposed method. The need to apply preprocessing on textual dataset is paramount important. Some of the importance of textual data preprocessing are described as follows.

The punctuation in a given language contains various forms of noise such as punctuation marks, emojis and emoticons, and non-standard text in a different form. Similarly, stop words removal is required as they are not content-bearing terms in a given sentence. Hence, they have to be removed from the text under preprocessing without causing any change in the meaning of the sentence using the natural language toolkit (NLTK) library [34,35]. To validate the proposed method, a three-class dataset with each class containing 1109, 771, and 8804 samples respectively.

3.1 Dataset description

The dataset was constructed by leveraging Twitter's search Application Programming Interface (API) using Python programming. It is evident that Python and its associated libraries are emerged and widely-used as they are flexible for conducting data analytics, particularly in the domain of machine learning. To ensure the relevance of the dataset to the intended application, hashtags associated with the three smart cities in Saudi Arabia, namely NEOM, Jeddah, and Riyadh, were used as search terms. Consequently, the most relevant hashtags including "#smart_cities", "#NEOM", "مدن مستدامة", "مدن ذكية", "مدينة ذكية", and "مدينة مستدامة", were extracted using inbuilt Python function.

The initial seed terms were carefully chosen to reflect the scope of the study and to ensure the suitability of the extracted data for subsequent analysis. In this study, the tweepy library was used to extract Arabic tweets from the Twitter website. Moreover, the official Twitter accounts of the target smart cities were included as search keywords to enhance the data collection step. The dataset is collected over a period spanning from January 2000 to 2023. This time period was deemed appropriate as it encompasses the reactions of people to the introduction of the Saudi smart cities' new index. It was recognized that the dataset would subsequently decrease in size due to the elimination of retweets and spam. The initial dataset comprised of 12000 tweets, which were later filtered and cleaned based on time-zone, location, and stratified random sampling. As a result, the dataset is reduced to 10,500 tweets, which formed the comprehensive dataset.

Given the targeted cities in this study are situated in KSA, an additional filtering process was conducted to identify tweets that originated from Saudi users based on their time-zone, and location information. However, it was observed that several tweets lacked location information in the user profiles. To overcome this limitation, a comprehensive list of Saudi Arabian city names, landmarks, and nicknames were utilized as supplementary labels for tweet user locations. Furthermore, a list obtained from a geographical database that includes information about millions of place names worldwide, including 25,253 place names for Saudi Arabia, was also employed following the approach of Mubarak and Darwish.

A longitudinal data set with seven columns and 12000 rows was collected. It provides information on user-generated content related to the topic of the study. The columns include the user name, user location, date, text, hashtags, and annotation. The "User_name" column includes the user name who generated the content, while the "User_location" column indicates the user's location. The "Date" column provides the date and time when the content was generated. The "Text" column contains the actual content generated by the user, while the "Hashtags" column lists any relevant hashtags used in the content. The "Annotation" column is a numerical value that indicates the class label of the content.

In general, the data provides a longitudinal view of user-generated content related to specific topics that are useful for analyzing trends and patterns over time. Despite the widespread availability of social media data to the public, concerns raised over the use of user profiling by businesses for commercial purposes. In this research work, Twitter was used as the main data source, where user phone numbers and addresses were not disclosed to ensure privacy. In an effort to safeguard user privacy, any phone numbers or names incorporated with the tweet content were removed during the data collection process. Furthermore, solely the tweet texts, time stamp, and location were collected, with

no other user-related information being extracted. These measures were implemented to uphold the ethical principles of data collection and prevent any potential infringement of user privacy rights. Table 2 presents a sample of Arabic tweets after undergoing preprocessing.

Table 2: Initial raw data sample after preprocessing

Tweet	Class
تشارك غدًا أمانة منطقة الرياض في معرض السعود.	1
مدينه المنورة او مكة والله اني ذكيه	3
هيئة الاتصالات والفضاء والتقنية توقع مذكرة تعاون للإستفادة البنية التحتية لقطاع المياة تمديد الألياف الضوئية سيساهم مضاعفة إنجاز وصول الخدمة للمستفيدين	1

The manual annotation process was conducted by language specialists on the dataset in order to reduce any potential annotator bias. To ensure accuracy, two separate volunteers were assigned to review the labeled data with the support of a manual method that help them in distinguishing between the various groups. The ultimate stage presents a new column that contains the classes of tweets.

3.1.1 Exploratory Data Analysis

Prior to conducting the sentiment analysis, it is imperative to perform a comprehensive analysis of the corpus to understand the data types that are essential in the model creation and prediction. Moreover, identify the features that stem from the corpus that contributes to the better performance of the proposed model. The exploratory data analysis was undertaken using sentence-based, and word-based features to facilitate data processing at multiple levels [36]. The analysis is performed using Natural Language Toolkit (NLTK) library through Python script.

To validate the proposed method, an Arabic Twitter dataset with three labels is employed as visualized in Figure 1. The number of samples in each class label are 1109, 771, and 8804 respectively.

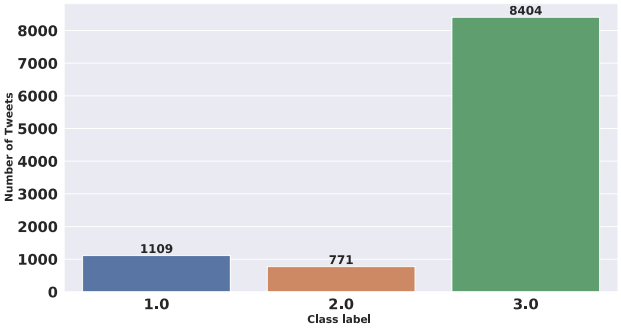


Fig. 1: Number of Tweets for each label.

Fig. 2 presents the most frequently words in the corpus along with their respective classes. Notably, the term "نيوم" appears to be heavily used; however, without further context, it is unclear whether it is used in a positive or negative category. Moreover, only one positive expression, "smart cities: المدن الذكية" was observed among these frequently occurring words. The word "5G: الخامس الجيل" was the most frequent term indicating its importance even though it remains uncertain at this stage whether its occurrence is positive or negative. To gain a deeper understanding of the usage patterns of these words, an investigation into their contextual usage was initiated by examining the most frequent bigram to provide a more comprehensive perspective of the data. The most frequent terms related to the topics examined were, unsurprisingly, "#smart_cities", "#NEOM", "نيوم", " الالياف الضوئية ", "مدينة ذكية ", " المدن ذكية", and "مدينة مستدامة".

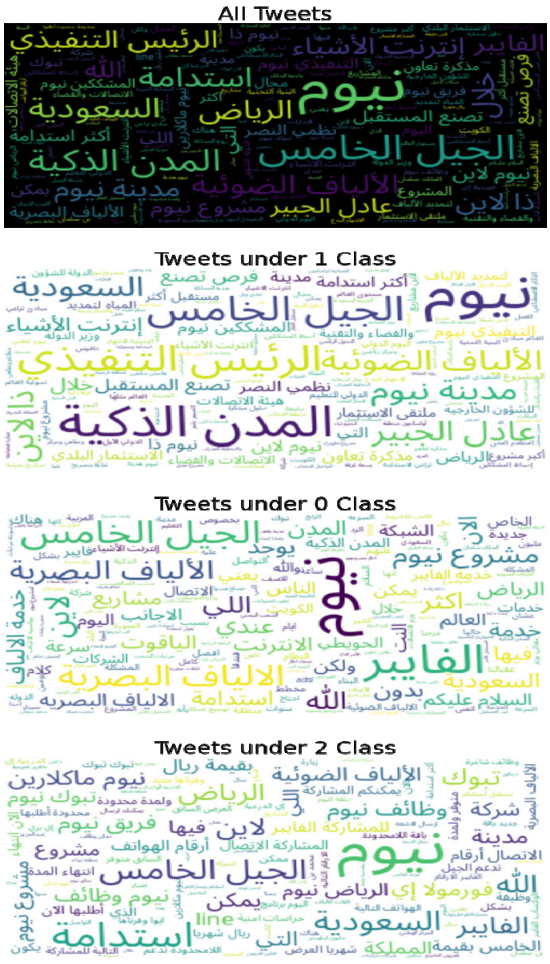


Fig. 2: Word cloud depicting the most frequent words appearing in all classes.

3.2 Tweets pre-processing

To improve the performance of the proposed model, the corpus passes through data pre-processing steps. The data preprocessing process underwent several interrelated and complementing steps, including the removal of spam and non-Arabic tweets, and the elimination of repetitive information such as retweets. To accomplish this, Tweets containing Uniform Resource Locators (URLs) were excluded. This is because prior studies have shown that tweets with URLs are often news articles or spam, which can introduce noise and bias into the classification process. Additionally, Tweets were filtered to include only those in the Arabic language. This is important because translation from other languages into Arabic can adversely affect the efficiency of the classifiers. Unnecessary features that might reduce classification accuracy were removed. This included user mentions (@user), numbers, special characters (e.g., @ # ~), and stop words (e.g., ",", ".", ";"). A Python script was utilized to automate the preprocessing tasks. This script efficiently removed the unwanted features and prepared the corpus for applying the classifiers.

To normalize and tokenize the tweet corpus, the NLTK library was used in Python script. While emoticons can convey sentiment, they were eliminated from the dataset due to prior research demonstrating that classifiers often confuse the parentheses in emoticons with those in quotations. Furthermore, research has shown that retaining emoticons during classification can decrease the performance of the model when working with Arabic tweets due to the right-to-left orientation of the Arabic language, which is reversed in emoticons.

In a text processing task, there are several interlinked and complementary steps including lexical analysis, removal of punctuation marks and stop words, and stemming among others [37]. The pre-processing steps are essential to transform words into numerical features that suit the specific machine-learning algorithm under consideration. It is evident that pre-processing steps in NLP depend on the domain in general and on the problem in particular [38]. Removal of punctuation is required to bring different forms of the same word into a single meaning otherwise, the same word can be interpreted and treated separately.

The punctuation in a given language contains noise in various forms such as punctuation, emotions, and text in a different form [39]. Similarly, stop words removal has to be done as they are not content-bearing terms. Hence, they have to be removed from the text under preprocessing without causing any change in the meaning of the sentence using NLTK library [40].

In this subsection, preprocessing of the training, validation, and testing is discussed. The data contains an imbalanced number of samples in the respective classes which pre-dominantly affects the effectiveness (accuracy) of the method unless it is preprocessed to handle the trade-off due to the imbalanced nature of the dataset [41,42]. To make this argument valid, prediction of negative, positive, and neutral tweets take place by splitting the training data and then compare the predicted labels to the human-judged labels. It is noted that data preprocessing such as cleaning from html and XML tags, white spaces, numbers, special characters, and stem the words under go as shown in Fig. 3.

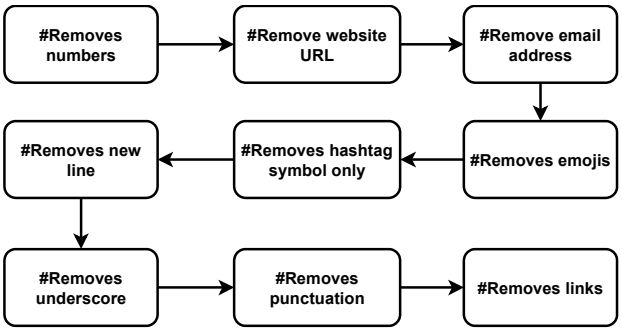


Fig. 3: Clean text steps.

To prepare the Arabic Tweets for effective analysis and classification, several additional preprocessing steps were performed: (1) Tokenization: Tweets were tokenized, which involves segmenting sentences into individual words [43, 44]. This is a fundamental step in NLP, as it allows for further analysis and feature extraction. (2) Normalization: Specifically for Arabic text, normalization was performed to unify certain types of Arabic letters that exist in different shapes [45]. This is important because these different shapes can represent the same letter, and failure to normalize them can introduce noise and inconsistency into the data [46,47].

As prior research has indicated that stemming algorithms are not effective for processing Dialectal Arabic words [48], such algorithms were not utilized in this study. The data collection, cleaning, filtering, and pre-processing procedures employed in the study are graphically depicted in Fig. 3. Word Embeddings are utilized to accurately depict the text in the dataset [49,50]. This approach provides a more effective representation of the text within the dataset when compared to conventional vectorization methods. Prior to utilizing Keras Embedding with the input corpus, it has to pass through tokenization and encoding the dataset into integers. To carry out this particular task, a tokenizer algorithm available in the Keras library is used. To train the Model, the list of tokenized words of varying length must be converted into a sequence of fixed length

[51]. To conclude this conversion, the character labels from the class column must be converted to integers for compatibility with the embedding layer.

Term Frequency-Inverse Document Frequency (TF-IDF) is a commonly used method in text mining to determine the importance of a term in a corpus. The term frequency (TF) measures the frequency of a term in a document, whereas the inverse document frequency (IDF) measures how rare or common the term is across all documents in the corpus. The product of these two values results in a score that indicates the relevance of the term to the document in which it occurs. This score can be used to rank documents based on their similarity to a query or to extract keywords from a document. TF-IDF has found applications in various domains such as search engines, recommender systems, and NLP [52].

3.3 BERT Model

BERT (Bidirectional Encoder Representations from Transformers) is a powerful language representation model developed by Google and released in 2018 [53, 54]. It is a deep neural network based on the Transformer architecture that uses unsupervised, self-supervised, and supervised learning to pre-train a large language model. BERT has been trained on a wide variety of domains and tasks, including question answering [55], natural language understanding, and inference [56]. BERT is a natural language processing tool essentially used in tasks that rely on understanding the ontological context of a sentence. The BERT utilizes a technique namely bidirectional encoding that allows it to look back and forth at the context of the whole sentence when making predictions or inferences. This makes BERT more effective and accurate than other ordinary language models that only look at the context of a single word or phrase once.

The BERT model is known for its effectiveness and efficiency with less time complexity. It can process large amounts of text data(corpus) in a few seconds. This makes it ideal for large-scale applications, such as search engines and question-answering systems. Hence, the success of BERT has led to the development of many other language models including GPT-3, which is based on the same architecture [57, 58]. BERT is already being used in many text-processing based applications including chatbots, question-answering systems, and text summarization.

BERT works by training the deep neural network algorithm on a large corpus of textual data [59]. The neural network learns to generate representations of words and phrases that are useful and content bearing terms for a particular task. For instance, if the task is to determine the sentiment of a sentence, then the representations learned by the BERT model are used to determine which words in the sentence have a positive or negative sentiment.

The development of BERT into being becomes major breakthroughs that gives momentum in the domain of NLP. It enabled and motivated researchers globally to engage and create more accurate and efficient models for natural language tasks in variants of languages. Hence, using the BERT model, many researchers across the world were able to develop better question-answering and automated text summarization systems.

BERT is founded on a Transformer (an attention mechanism that learns the contextual relationships between words in a text) [54]. A primary Transformer consists of an encoder to read the text input and a decoder to generate an output for the task. Since BERT's purpose is to design a language representation model, it only requires the encoder element. The input for the encoder for BERT creates sequences of vectorized tokens (words) before processed in the neural network [60]. It has to be noted that the BERT needs extra metadata that needs to be included in its input: (1) Token embeddings: whereby a [CLS] token is appended at the beginning of the first sentence and a [SEP] token is inserted at the end of each sentence, (2) Segment embeddings: markers indicating sentence A or sentence B are added with each token so that encoder could differentiate between sentences, and (3) Positional embeddings: a positional embedding (index) is attached to each token indicating their respective positions in the sentence.

Essentially, the transformer utilizes a layer that maps sentences to sequences; thus, resulting in a list of words (vectors) with a 1:1 correspondence between input and output tokens in the same index [61]. BERT does not developed to predict the subsequent word in the sentence. Rather, the BERT uses two strategies namely masked LM and next sentence prediction [62, 63].

3.4 The proposed BERT model

BERT is a state-of-the-art model based on transformers developed by Google. In this research work, the BERT model is fine-tuned for the classification of Arabic tweets for sentiment analysis of smart city related tweets. The main steps of the BERT model are presented in Fig. 4.

The steps of a machine learning model generally involve breaking down the model into its constituent parts. The basic outline of the steps involved in a typical machine learning model are listed as below.

1. Data preprocessing: The first step is to prepare the data for analysis. This involves tasks such as cleaning the data, removing irrelevant features such as special characters, URLs, and hyperlinks, stop-words removal, and handling missing data.
2. Feature selection: The relevant features are selected from the dataset. This is important to reduce the computational complexity. In this work, the relevant vectorized terms get counted and selected, compute the relevance of a term in terms of TF-IDF distribution.
3. Model selection: Selecting a model that fits well to the data is a crucial step in machine learning. There are different types of models such as regression, decision trees, neural networks, etc. Every model has specific strengths and weaknesses.
4. Model training: Once the model is selected, it is trained on the data set. The model learns the patterns and relationships in the data by adjusting its parameters. This is typically done using an optimization algorithm back and forth several times until the most optimized model is obtained.
5. Model evaluation: After the model is trained, it is evaluated on a test set data to measure its performance. To measure the performance of the proposed model, there are common evaluation metrics that can be used as benchmarks such as accuracy, Precision, Recall, F1, and ROC curve.
6. Hyperparameter tuning: Tuning the model hyperparameters is the process of selecting the best set of values that optimize the model's accuracy. This is usually done through a combination of trial and error and automated optimization techniques.

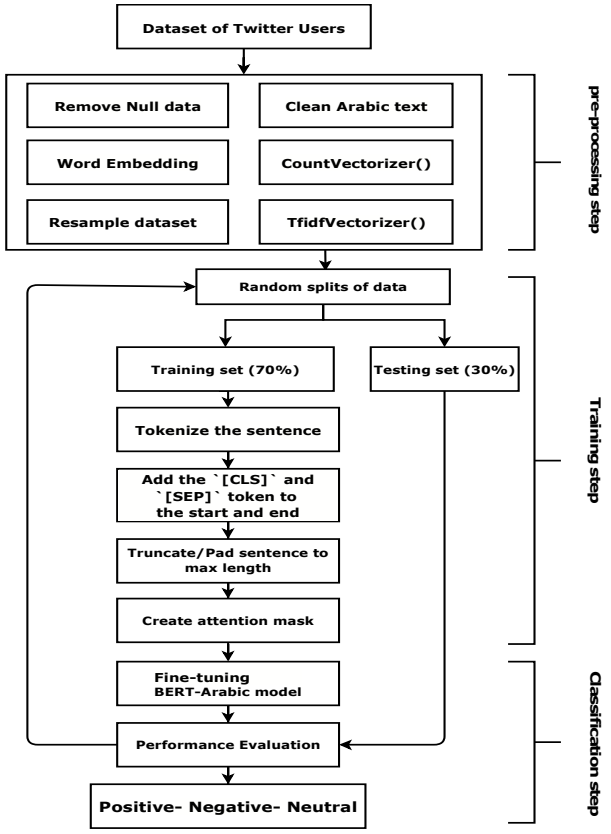


Fig. 4: The main steps of the BERT model.

4. Evaluation metrics

The common performance metrics are used for evaluating the machine learning models under investigation. The proposed framework can be evaluated using several metrics, such as accuracy, F1-score, recall, precision, ROC curve, and AUC. To evaluate the performance of a machine learning model, considering an imbalanced class dataset, instead of sticking to the accuracy of the model only, it is ideal to consider a precision, recall, and another harmonic measure such as the F1-Measure are better suited to evaluating the performance of the classifier.

The binary classification problem only requires classification between two kinds of outputs, positive and negative. This means that True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are the four outcomes that can really occur [64]. The accuracy, precision, recall and F1 are calculated as per equations 1-4 respectively [65]:

Accuracy is defined as the proportion of true positives and true negatives to total predictions.

$$\text{Accuracy} = (TP + TN)/ (TP + FP + FN + TN) \quad (1)$$

Precision is the proportion of true positives to all predicted positives. The precision is formulated as follows:

$$\text{Precision} = (TP)/(TP + FP) \quad (2)$$

The recall, also known as the sensitivity measure, is the proportion of true positives to all actual positives and false negatives.

$$\text{Recall} = (TP)/(TP + FN) \quad (3)$$

The F1-score is the harmonic mean of precision and recall.

$$F1 = 2\ (\text{precision} * \text{recall})/(\text{precision} + \text{recall}) \quad (4)$$

The confusion matrix is typically useful tool in the field of machine learning to measure and visualize the performance of a classification algorithm. It is a tabular representation that summarizes the performance of the model. The matrix displays the number of correct and incorrect predictions made by a model and classifies them into four different categories: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) [66]. The confusion matrix is usually created by interpreting the output of a classification algorithm and then comparing it to the actual labels of the dataset. The matrix then displays the number of TP, FP, FN and TN. A true positive is when the model has correctly identified an item as belonging to a certain class. A false positive is when the model incorrectly identifies an item as belonging to a certain class. A true negative is when the model correctly identifies an item as not belonging to a certain class. And a false negative is when the model incorrectly identifies an item as not belonging to a certain class.

5. Experimental Results and Analysis

This section explores the use of Twitter mining to predict customer satisfaction for Saudi smart cities, thereby addressing the research question and achieving the research objective. Various baseline and state-of-the-art models are compared and enhanced to achieve the research objective. The study begins with machine learning models, with different embeddings, and transformer networks, including the AraBERT model. This allows for a comprehensive evaluation of the impact of Twitter features on customer satisfaction prediction.

The AraBERT model was found to be highly accurate in predicting customer satisfaction based on tweets, and outperformed other models. The findings of this study contribute to the understanding of customer satisfaction in smart cities and the role of Twitter mining in predicting such satisfaction. The proposed framework is implemented using Python programming language and associated libraries and packages. Generally, the programming language, operating system, RAM, and CPU capacity are detailed and the configuration is presented in Table 3.

Table 3: Experimental environment setup.

Parameter	Value
Programming Language	Python 3.6.13
OS	Windows 11 Pro
RAM	32GB
CPU	Intel ® Xeon ® W-1250 @ 3.30 GHz

"

5.1 Machine learning with SMOTE

"

The present study investigates several methods for balancing imbalanced training data sets, including under-sampling, over-sampling, and combining both over and under-sampling. Under-sampling technique is utilized to balance data sets by minimizing the size of the majority class when the data quantity is sufficient. In such cases, all samples in the minority class are retained, and an equal number of samples are randomly chosen from the majority class. However, when there is a significant disparity between the positive and negative tweets, as in our model, which only contains 771 negative tweets, under-sampling may not be particularly effective.

To address the challenge of data imbalance in predicting customer satisfaction, this study explores various resampling scenarios: (1) Over-Sampling: When the size of data is insufficient, over-sampling techniques can be employed to balance the dataset by increasing the size of minority samples. This can be achieved using methods such as SMOTE, or Adaptive Synthetic Sampling (ADASYN). (2) Under-Sampling: This technique can be used to decrease the size of the majority class, thereby balancing the dataset. Common under-sampling methods include random under-sampling, and cluster-based under-sampling. (3) Combined Sampling: Another approach is to combine both under and over-sampling methods to achieve a more balanced dataset.

In this study, the three resampling methods (over-sampling, under-sampling, and combined sampling) are evaluated for their effectiveness in the given model. This evaluation helps determine the most suitable resampling technique for predicting customer satisfaction in the context of smart cities as shown in Table 4.

Table 4: Comparison of Baseline and SMOTE-based Machine Learning Models.

SMOTE	Classifier	Accuracy	Precision	Recall	F1
SMOTE	LR	0.8931	0.8902	0.8931	0.8907
	Linear SVC	0.8986	0.8925	0.8986	0.8910
	Extra Trees	0.8992	0.8929	0.8992	0.8917
	Random Forest	0.8986	0.8920	0.8986	0.8902
	Ada Boost	0.8519	0.8487	0.8519	0.8498
	Gradient Boosting	0.8574	0.8731	0.8574	0.8637
	Voting	0.8950	0.8921	0.8950	0.8923
	Stacking	0.8697	0.8703	0.8697	0.8698
as el	LR	0.8820	0.8843	0.8820	0.8582
	Linear SVC	0.8983	0.8953	0.8983	0.8848

	Extra Trees	0.8918	0.8891	0.8918	0.8741
	Random Forest	0.8889	0.8873	0.8889	0.8685
	Ada Boost	0.8496	0.8295	0.8496	0.8271
	Gradient Boosting	0.8837	0.8761	0.8837	0.8649
	Voting	0.8872	0.8865	0.8872	0.8665
	Stacking	0.8717	0.8610	0.8717	0.8637
Borderline SMOTE	LR	0.8940	0.8897	0.8940	0.8910
	Linear SVC	0.9002	0.8946	0.9002	0.8926
	Extra Trees	0.9005	0.8942	0.9005	0.8920
	Random Forest	0.9005	0.8943	0.9005	0.8911
	Ada Boost	0.8276	0.8492	0.8276	0.8355
	Gradient Boosting	0.8558	0.8703	0.8558	0.8617
	Voting	0.8960	0.8921	0.8960	0.8932
	Stacking	0.8623	0.8619	0.8623	0.8620
ADASYN	LR	0.8960	0.8937	0.8960	0.8945
	Linear SVC	0.8979	0.8915	0.8979	0.8903
	Extra Trees	0.8976	0.8909	0.8976	0.8903
	Random Forest	0.8970	0.8900	0.8970	0.8899
	Ada Boost	0.8989	0.8924	0.8989	0.8903
	Gradient Boosting	0.8347	0.8525	0.8347	0.8409
	Voting	0.8516	0.8708	0.8516	0.8591
	Stacking	0.8947	0.8935	0.8947	0.8937
Tomek Links	LR	0.8548	0.8603	0.8548	0.8573
	LR	0.8817	0.8840	0.8817	0.8578
	Linear SVC	0.8973	0.8943	0.8973	0.8838
	Extra Trees	0.8927	0.8892	0.8927	0.8760
	Random Forest	0.8882	0.8862	0.8882	0.8683
	Ada Boost	0.8519	0.8418	0.8519	0.8425
	Gradient Boosting	0.8856	0.8792	0.8856	0.8677
	Voting	0.8866	0.8869	0.8866	0.8653
	Stacking	0.8684	0.8588	0.8684	0.8620
Near Miss	LR	0.7978	0.8760	0.7978	0.8201
	Linear SVC	0.8056	0.8795	0.8056	0.8269
	Extra Trees	0.5774	0.8060	0.5774	0.6247
	Random Forest	0.5752	0.8081	0.5752	0.6221
	Ada Boost	0.3483	0.7533	0.3483	0.3807
	Gradient Boosting	0.5677	0.8362	0.5677	0.6173
	Voting	0.3846	0.8081	0.3846	0.4059
	Stacking	0.3296	0.7695	0.3296	0.3465
Random Over Sampler	LR	0.8947	0.8932	0.8947	0.8935
	Linear SVC	0.8989	0.8927	0.8989	0.8913
	Extra Trees	0.8918	0.8867	0.8918	0.8763
	Random Forest	0.8911	0.8857	0.8911	0.8859
	Ada Boost	0.7638	0.8230	0.7638	0.7842
	Gradient Boosting	0.8364	0.8734	0.8364	0.8490
	Voting	0.8898	0.8904	0.8898	0.8897
	Stacking	0.8561	0.8659	0.8561	0.8604
Random Under Sampler	LR	0.7871	0.8752	0.7871	0.8140
	Linear SVC	0.7887	0.8755	0.7887	0.8156
	Extra Trees	0.7239	0.8781	0.7239	0.7707
	Random Forest	0.7531	0.8729	0.7531	0.7907
	Ada Boost	0.7680	0.8437	0.7680	0.7938
	Gradient Boosting	0.7855	0.8548	0.7855	0.8071
	Voting	0.7946	0.8740	0.7946	0.8191
	Stacking	0.7268	0.8445	0.7268	0.7604
Edited Nearest Neighbours	LR	0.8620	0.8657	0.8620	0.8346
	Linear SVC	0.8746	0.8742	0.8746	0.8566
	Extra Trees	0.8503	0.8483	0.8503	0.8033
	Random Forest	0.8490	0.8472	0.8490	0.8014
	Ada Boost	0.8205	0.8099	0.8205	0.7547
	Gradient Boosting	0.8526	0.8444	0.8526	0.8129
	Voting	0.8500	0.8503	0.8500	0.8033
	Stacking	0.8506	0.8430	0.8506	0.8059

The table includes eight classifiers, namely Logistic Regression, LinearSVC, Random Forest classifier, Extra Trees classifier, AdaBoost classifier, Gradient Boosting classifier, Voting classifier, and Stacking classifier. The models are evaluated using the performance metrics namely accuracy, recall, precision, and F1 score. Overall, the classifiers achieved a high level of accuracy ranging from 0.849 to 0.898. LinearSVC had the highest performance, whereas AdaBoost classifier had the lowest accuracy, precision, and F1 score. The results indicate that SMOTE technique can enhance the performance of classifiers in predicting imbalanced data.

Furthermore, the experimental results of the Baseline sampling technique are provided in Table 4. Accordingly, several machine learning models using the Borderline SMOTE technique are presented in Table 4. The experimental result depicted in Table 4 demonstrates that all models achieved better accuracy scores, with Linear SVC, Extra Trees classifier, and Random Forest classifier achieving the highest accuracy scores of 0.8983, 0.8918 and 0.8889 respectively. Moreover, the precision and recall scores of all models were close to their accuracy scores, indicating that the models were performing well in terms of classifying the different classes. Among the models, Linear SVC, Extra Trees classifier, and Random Forest classifier had the highest precision and recall scores. On the other hand, the AdaBoost classifier scored lesser accuracy compared with the other models in classifying the different classes.

The results of ADASYN of machine learning models are provided in Table 4. Table 4 presents the performance of several machine learning models trained on a dataset generated using the ADASYN oversampling technique. The results show that the highest accuracy was achieved by the Random Forest model with a score of 0.8988, followed closely by the LinearSVC model with a score of 0.8979. All the models in the table performed well with an accuracy of over 0.83, indicating that the ADASYN technique was effective in generating synthetic data that improved the models' performance. In terms of precision, the ExtraTrees classifier with 0.8908 had the highest precision score, followed by the Random Forest model with a score of 0.8924. However, the LogisticRegression model had the highest recall score of 0.8959. The F1 score was highest for the Logistic Regression model, with a score of 0.8945.

The results of Tomek Links resampling technique are provided in Table 4. The table provides the results of various machine learning models with Tomek Links. For the TomekLinks undersampling technique, all the models achieved good perfor mance with accuracy scores ranging from 0.8519 to 0.8972. The Linear SVC model achieved the highest accuracy score of 0.8972, followed by ExtraTrees

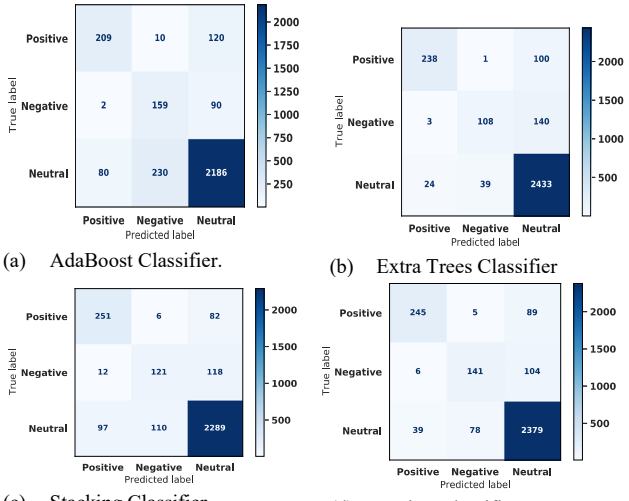
model (0.8927), Random-Forest Classifier (0.8882), and Gradient Boosting model (0.8856). The AdaBoost model achieved the lowest accuracy score of 0.8519. In terms of precision, all models achieved scores above 0.84, except for AdaBoost model which had a precision score of 0.8418. The Linear SVC model had the highest precision score of 0.8943. The recall scores ranged from 0.8817 to 0.8972, with the Linear SVC model achieving the highest score. The F1 scores ranged from 0.8578 to 0.8838, with the Linear SVC model achieving the highest score.

For the Near Miss undersampling technique, the performance of the models was generally lower than that achieved with Tomek Links. The Logistic Regression model achieved the highest accuracy score of 0.7978, followed by Linear SVC (0.8056), Gradient Boosting model (0.5677), and ExtraTrees Classifier (0.5774). The AdaBoost model achieved the lowest accuracy score of 0.3483. In terms of precision, all models achieved scores above 0.75, except for Voting classifier (0.8081) and Stacking Classifier (0.7695). The Linear SVC model had the highest precision score of 0.8795. The recall scores ranged from 0.3296 to 0.8056, with the Linear SVC model achieving the highest score. The F1 scores ranged from 0.3465 to 0.8269, with the Linear SVC model achieving the highest score.

The results of Random Over Sampler sampling technique are provided in Table 4. Table 4 presents the results of machine learning models trained using the Random Over Sampler method. The table reports accuracy, precision, recall, and F1 metrics for eight different classifiers. The results show that the Linear SVC model achieved the highest accuracy of 0.8989, followed by the Logistic Regression classifier with 0.8947 accuracy. On the other hand, AdaBoost model has the lowest accuracy of 0.7638. Regarding precision, recall, and F1 score, the Logistic Regression model and the Voting Classifier have achieved the highest values in these metrics for most of the classifiers.

The results of Edited Nearest Neighbours of machine learning models are provided in Table 4. The presented table shows the results of various machine learning models applied on a dataset using the Edited Nearest Neighbours (ENN) sampling technique. The results of each model are presented in the table with the corresponding performance metrics. The table indicates that the Linear SVC model achieved the highest accuracy (87.46%), followed by Logistic Regression with an accuracy of 86.20%. The ExtraTrees and Random Forest models achieved accuracy scores of 85.02% and 84.90%, respectively. In terms of precision, the Linear SVC model again achieved the highest score (87.41%), while Logistic Regression achieved a precision score of 86.57%. The models with he highest recall were the Linear SVC (87.46%) and Logistic Regression (86.20%). The F1 score, which balances the precision and recall of a model, was highest for Linear SVC (85.66%), followed by LogisticRegression with a score of 83.46%.

The experimental results using the confusion matrix for the machine learning models with borderline SMOTE are presented in Figure 5. The confusion matrix for a Random Forest model applied to Arabic tweets can provide a numerical analysis of the model's performance. The matrix would contain four values: true positive, false positive, true negative, and false negative, which represent the number of tweets that were correctly and incorrectly classified by the model. Suppose a Random Forest model applied to a set of Arabic tweets may have correctly classified 1200 positive tweets (true positive), 800 negative tweets (true negative), misclassified 200 tweets as positive when they were actually negative (false positive), and misclassified 100 tweets as negative when they were actually positive (false negative).



(a) AdaBoost Classifier. (b) Extra Trees Classifier

(c) Stacking Classifier (d) Voting Classifier

Fig. 5: The confusion matrix of Machine learning models with Borderline SMOTE.

In addition, the area under the ROC curve (AUC) is used to measure the performance of the proposed model. In a machine learning algorithm, the ROC curve demonstrates the performance of a model in terms of its true positive rate (TPR) and false positive rate (FPR). The ROC curve is created by plotting TPR against FPR at various threshold settings. The closer the ROC curve is to the upper left corner, the better the model is at predicting positive outcomes. A perfect model would have ROC curve that looks like a straight line from the lower left corner to the upper right corner. In practice, most models have a curved ROC curve that oscillates between the two corners. The

ROC curve of machine learning models with borderline SMOTE is provided in Figure 6 for AdaBoost, Extra Trees, Stacking and Voting classifiers.

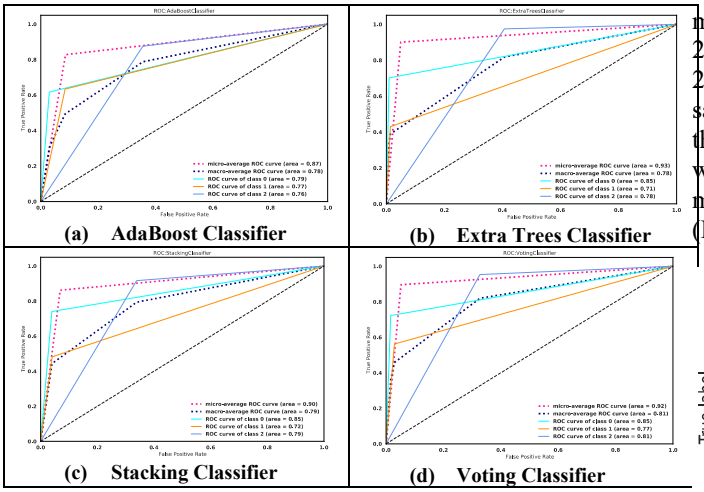


Fig. 6: The ROC curve of machine learning models with Borderline SMOTE.

5.2 Deep learning (Arabic BERT)

In this section, experimental results of the BERT classifier are presented and discussed. The classification report of the BERT model is presented in Table 5. The results are presented in terms of the performance evaluation metrics such as precision, recall, F1-score, and support of three classes namely Positive, Negative, and Neutral. Moreover, Table 5 shows that the BERT model achieved a precision score of 0.77 for the Positive class, 0.62 for the Negative class, and 0.93 for the Neutral class. The recall score was 0.74 for the Positive class, 0.56 for the Negative class, and 0.94 for the Neutral class. The F1-score was 0.76 for the Positive class, 0.59 for the Negative class, and 0.93 for the Neutral class. The overall accuracy of the proposed method is profoundly determined to be 0.89.

Table 5: The classification report of the BERT Classifier.

	Precision	Recall	F1-Score	Support
Positive	0.77	0.74	0.76	339
Negative	0.62	0.56	0.59	251
Neutral	0.93	0.94	0.93	2496
accuracy			0.89	3086
Macro avg	0.77	0.75	0.76	3086
Weighted avg	0.88	0.89	0.89	3086

Table 5 shows that the macro-average F1-score is 0.76, while the weighted average F1-score was 0.89. The support measure provides the number of instances in each class, whereby 339 instances belong to the positive class, 251 instances goes to the Negative class, and 2496 instances are from the Neutral in the dataset used to evaluate the BERT Classifier.

The confusion matrix for the BERT model applied to Arabic tweets is given in Figure 7. For instance, a BERT model applied to a set of Arabic tweets correctly classified 255 positive tweets (TP), 149 negative tweets (TN), and 2334 tweets correctly classified as neutral. The remaining samples of the diagonal line are misclassified. For instance, the 13 and 71 tweets are misclassified as positive when they were actually negative (FP), and 5 and 97 tweets are misclassified as negative when they were actually positive (FN).

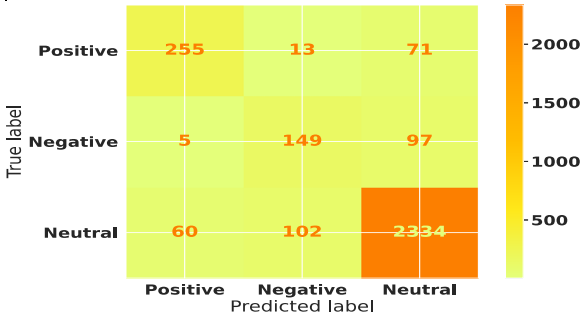


Fig. 7: The confusion matrix of BERT Classifier.

The ROC curve of the proposed BERT models is presented in Figure 8. According to the experimental results, the ROC curves for positive, negative, and neutral classes are found to be 0.86, 0.78, and 0.83 respectively with weighted micro-average and macro-average experimental results for three of the classes are found to be 0.92 and 0.82 respectively.

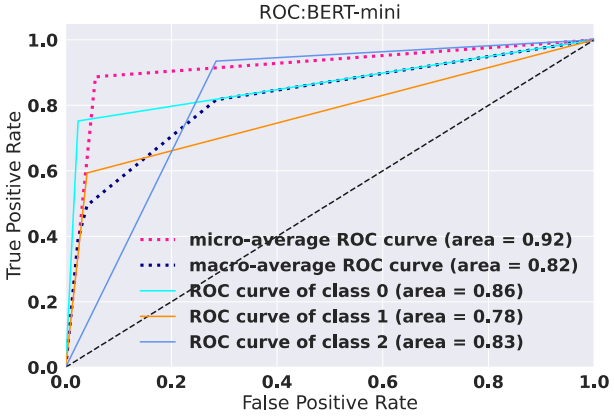


Fig. 8: The confusion matrix of BERT Classifier.

5.3 Comparison between the models

The Random over-sampler technique works by duplicating the number of samples of minority class to over-sample it, whereas ADASYN and SMOTE [67, 68] generate new synthetic samples in the dataset using interpolation. There is a difference in the types of samples

used for generating new synthetic data. The ADASYN algorithm aims to address the issue of imbalanced datasets by specifically generating new samples near the original samples that are misclassified by a k-Nearest Neighbors classifier. These misclassified samples, often referred to as hard-to-learn samples, are given priority in the generation process. In contrast, the standard implementation of SMOTE does not differentiate between easy and hard-to-learn samples when utilizing the nearest neighbors rule for classification.

It should be noted that SMOTE may connect inliers and outliers, hence this approach provides three additional options for generating samples. Both ADASYN and SMOTE methods focus on samples that lie near the decision boundary of the optimal classifier. These methods generate new samples in the direction opposite to the class of their nearest neighbors.

Table 6 depicts the experimental results of the comparison between machine learning models on the Arabic tweets. The table provides the results of a comparison between various machine learning models using different sampling techniques. The first and second columns in Table 6 represents the sampling technique and type of classifier used in the respective experiment. The experimental results show that the sampling techniques, such as SMOTE, Borderline SMOTE, and SMOTE-Tomek, produce higher accuracy, precision, recall, and F1 scores when comparison with the other sampling techniques, such as NearMiss, Random Under Sampler, and Random Over Sampler. In addition, the BERT classifier achieved better performance with a balanced accuracy.

Table 6: The results of comparison between machine learning models.

SMOTE	Model	Accuracy	Precision	Recall	F1
SMOTE	ExtraTrees	0.8992	0.8929	0.8992	0.8917
EditedNearestNeighbours	LinearSVC	0.8746	0.8742	0.8746	0.8566
BorderlineSMOTE	RandomForest	0.9005	0.8943	0.9005	0.8911
ADASYN	RandomForest	0.8989	0.8924	0.8989	0.8903
TomekLinks	LinearSVC	0.8973	0.8943	0.8973	0.8838
NearMiss	LinearSVC	0.8056	0.8795	0.8056	0.8269
RandomOverSampler	LogisticRegression	0.8947	0.8932	0.8947	0.8935
RandomUnderSampler	Voting Classifier	0.7946	0.8740	0.7946	0.8191
SMOTETomek	VotingClassifier	0.8992	0.8966	0.8992	0.8972
SMOTEENN	RandomForest	0.8412	0.8865	0.8412	0.8564
-	BERT	0.8872	0.8879	0.8872	0.8872

In this study, the sampling methods used are Edited Nearest Neighbour, Borderline SMOTE, ADASYN, TomekLinks, Near Miss, Random Over Sampler, Random Under Sampler, SMOTETomek, and SMOTEENN. Along with the specified sampling techniques (SMOTE), the following classifiers including Extra tree, Linear SVM, Random forest, Logistic regression, and random voting methods were used.

The SMOTE sampler was used with Extra Trees Classifier to score an accuracy, precision, recall, and F1-score of 0.8992, 0.8929, 0.8992, and 0.8917 respectively. Similarly, the Edited Nearest Neighbours sampling method was used along with the linear SVM and equal accuracy, precision, and recall of 0.8746 with F1-score 0. 8566.

Based on the comparative experimentation, the model with the highest accuracy was the Borderline SMOTE sampler and Random Forest classifier that achieving scores of 0.9005, 0.8943, 0.9005, and 0.8911, for accuracy, precision, recall, and F1-score. Other models, such as Near Miss with Linear SVC and Random Under Sampler with Voting classifier, registered relatively lower scores in the four metrics, indicating poorer performance. It is also worth noting that the BERT classifier achieved a relatively high accuracy score of 0.8872 as shown in Table 6.

6. Conclusion and Future work

In this study, a classification system for Arabic tweets based on the Arabic BERT model is proposed. The proposed system is applicable for sentiment analysis on Arabic tweets namely the Arabic BERT-based classification system. To validate the proposed method, several performance methods are used on a dataset consisting of labeled Arabic tweets from the Arabic Twitter Corpus. During validation, the dataset was divided randomly into training and test sets. The experimental results of the proposed method showed an accuracy of 89%, which is a promising result in light of the difficulty of dealing with Arabic language textual data analytics. The Arabic BERT-based classification system was able to accurately and reliably classify Arabic tweets into different sentiment categories. The experimental results of the proposed method demonstrated the effectiveness of using the Arabic BERT model in such tasks, and show that it is a viable option for the classification of Arabic language data that pave ways for future directions. Moreover, the experimental results of this research work are encouraging and suggest that the system can be applied in other areas of sentiment analysis on Arabic language data. In this research work, a classification framework for Arabic tweets based on the Arabic BERT model is proposed. The study showed that the proposed approach is able to accurately classify tweets into alternative classes namely positive, negative, and neutral sentiments. The performance of the proposed model is evaluated using several well established and commonly used performance metrics including precision, recall, F1-score, and accuracy. The experimental results depicted that the proposed model achieved an accuracy of 88.7% in classifying Arabic tweets with a precision, recall, and F1-score rate of 88.7%, 88.7%, and 89% respectively. In this research work, promising experimental results were obtained for sentiment analysis on Arabic language data using different sampling methods over a number of classifiers. However, it still signifies that the performance of proposed work cab be farther improve using other

methods. One possible direction for future work is to improve the accuracy of the model, by incorporating additional features into the model and exploring other techniques of data pre-processing and filtering methods. It can be interesting to explore the use of other models such as LSTMs and GRUs, for sentiment analysis on Arabic language data. Applying transfer learning for sentiment analysis on Arabic language datasets is another potential.

References

[1] Ritchie, H.; Roser, M. Urbanization. Our world in data 2018.

[2] Eremia, M.; Toma, L.; Sanduleac, M. The smart city concept in the 21st century. Procedia Engineering, 181, pp. 12–19, 2017.

[3] Washburn, D.; Sindhu, U.; Balaouras, S.; Dines, R.A.; Hayes, N.; Nelson, L.E. Helping CIOs understand "smart city" initiatives. Growth, 17, pp. 1–17, 2009.

[4] Úbeda, R. ITU transforming cities in smarter and more sustainable. In Proceedings of the Third Meeting of the United for Smart Sustainable Cities Initiative, 26 Apr, Malaga, Spain, by ITU and UNECE, 2018.

[5] Nam, T.; Pardo, T.A. Conceptualizing smart city with dimensions of technology, people, and institutions. In Proceedings of the Proceedings of the 12th annual international digital government research conference: digital government innovation in challenging times, pp. 282–291, 2011.

[6] Moustaka, V.; Vakali, A.; Anthopoulos, L.G. A systematic review for smart city data analytics. ACM Computing Surveys (cSuR), vol 51, pp. 1–41, 2018.

[7] Gundecha, P.; Liu, H. Mining social media: a brief introduction. New directions in informatics, optimization, logistics, and production, pp. 1–17, 2012.

[8] Liu, B. Sentiment analysis and opinion mining. Synthesis lectures on human language technologies, vol 5, pp. 1–167, 2012.

[9] Alotaibi, S.; Mehmood, R.; Katib, I. Sentiment analysis of Arabic tweets in smart cities: A review of Saudi dialect. In Proceedings of the 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC). IEEE, pp. 330–335, 2019.

[10] S. Dixon, P.; 6, O. Twitter global mdau 2022, 2022.

[11] Sayce, D. The number of tweets per day in 2022, 2022.

[12] blogger, G. Saudi Arabia (KSA) population statistics [2022 updated]: GMI, 2022.

[13] Khorsheed, M.S. Saudi Arabia: from oil kingdom to knowledge-based economy. Middle East Policy, vol. 22, pp. 147–157, 2015.

[14] Smart City Observatory, I. Smart City Index 2021, 2021.

[15] Global media insight. SAUDI ARABIA SOCIAL MEDIA STATISTICS 2022, 2022.

[16] Sahu, T.P.; Khandekar, S. A Machine Learning-Based Lexicon Approach for Sentiment Analysis. In Research Anthology on Implementing Sentiment Analysis Across Multiple Disciplines; IGI Global, pp. 836–851, 2022.

[17] Steven, C.; Wella, W. The right sentiment analysis method of Indonesian tourism in social media Twitter. IJNMT (International Journal of New Media Technology), vol 7, pp. 102–110, 2020.

[18] Abdullah, B.; Alosaimi, N.; Almotiri, S. Reputation Measurement based on a Hybrid Sentiment Analysis Approach for Saudi Telecom Companies. International Journal of Advanced Computer Science and Applications, vol 12, 2021.

[19] Alotaibi, S.; Mehmood, R.; Katib, I. Sentiment Analysis of Arabic Tweets in Smart Cities: A Review of Saudi Dialect. In Proceedings of the 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC). IEEE, 2019.

[20] Hassonah, M.A.; Al-Sayyed, R.; Rodan, A.; Ala'M, A.Z.; Aljarah, I.; Faris, H. An efficient hybrid filter and evolutionary wrapper approach for sentiment analysis of various topics on Twitter. Knowledge-Based Systems 2020, 192, 105353.

[21] Annett, M.; Kondrak, G. A comparison of sentiment analysis techniques: Polarizing movie blogs. In Proceedings of the Conference of the Canadian Society for Computational Studies of Intelligence. Springer, pp. 25–35, 2008.

[22] Salas, A.; Georgakis, P.; Nwagboso, C.; Ammari, A.; Petalas, I. Traffic event detection framework using social media. In Proceedings of the 2017 IEEE International Conference on Smart Grid and Smart Cities (ICSGSC). IEEE, 2017, pp. 303–307.

[23] Li, M.; Ch'ng, E.; Chong, A.; See, S. The new eye of smart city: novel citizen sentiment analysis in twitter. In Proceedings of the 2016 International Conference on Audio, Language and Image Processing (ICALIP). IEEE, 2016, pp. 557–562.

[24] Al Nuaimi, A.; Al Shamsi, A.; Al Shamsi, A.; Badidi, E. Social media analytics for sentiment analysis and event detection in smart cities. In Proceedings of the Proc. 4th Int. Conf. Natural Lang. Comput.(NATL), 2018, pp. 57–64.

[25] Musto, C.; Semeraro, G.; de Gemmis, M.; Lops, P. Developing smart cities services through semantic analysis of social streams. In Proceedings of the Proceedings of the 24th International Conference on World Wide Web, 2015, pp. 1401–1406.

[26] Hasan, A.; Moin, S.; Karim, A.; Shamshirband, S. Machine Learning-Based Sentiment Analysis for Twitter Accounts. Mathematical and Computational Applications 2018, 23, 11.

[27] Manguri, K.H.; Ramadhan, R.N.; Amin, P.R.M. Twitter sentiment analysis on worldwide COVID-19

outbreaks. Kurdistan Journal of Applied Research 2020, pp. 54–65.
[28] Arambepola, N. Analysing the tweets about distance learning during COVID-19 pandemic using sentiment analysis. In Proceedings of the Proc of the International Conference on Advances in Computing and Technology, 2020, pp. 169–171.
[29] Alayba, A.M.; Palade, V.; England, M.; Iqbal, R. Arabic language sentiment analysis on health services. In Proceedings of the 2017 1st international workshop on arabic script analysis and recognition (asar). IEEE, 2017, pp. 114–118.
[30] Alomari, K.M.; ElSherif, H.M.; Shaalan, K. Arabic tweets sentimental analysis using machine learning. In Proceedings of the Advances in Artificial Intelligence: From Theory to Practice: 30th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2017, Arras, France, June 27-30, 2017, Proceedings, Part I 30. Springer, 2017, pp. 602–610.
[31] Aljameel, S.S.; Alabbad, D.A.; Alzahrani, N.A.; Alqarni, S.M.; Alamoudi, F.A.; Babili, L.M.; Aljaafary, S.K.; Alshamrani, F.M. A Sentiment Analysis Approach to Predict an Individual's Awareness of the Precautionary Procedures to Prevent COVID-19 Outbreaks in Saudi Arabia. International Journal of Environmental Research and Public Health 2020, 18, 218.
[32] Al-Hashedi, A.; Al-Fuhaidi, B.; Mohsen, A.M.; Ali, Y.; Gamal Al-Kaf, H.A.; Al-Sorori, W.; Maqtary, N. Ensemble classifiers for arabic sentiment analysis of social network (twitter data) towards covid-19-related conspiracy theories. Applied Computational Intelligence and Soft Computing 2022, 2022, 1–10.
[33] Sarkar, D. Python for Natural Language Processing. In Text Analytics with Python; Apress, 2019; pp. 69–114.
[34] Wang, M.; Hu, F. The Application of NLTK Library for Python Natural Language Processing in Corpus Research. Theory and Practice in Language Studies 2021, 11, 1041–1049. https://doi.org/10.17507/tpls.1109.09.
[35] Jiwani, N.; Gupta, K.; Whig, P. Analysis of the Potential Impact of Omicron Crises Using NLTK (Natural Language Toolkit). In Proceedings of Third Doctoral Symposium on Computational Intelligence; Springer Nature Singapore, 2022; pp. 445–454.
[36] Soler-Company, J.; Wanner, L. On the role of syntactic dependencies and discourse relations for author and gender identification. Pattern Recognition Letters 2018, 105, 87–95.
[37] II, T.B. Applied Natural Language Processing with Python; Apress, 2018.
[38] Kulkarni, A.; Shivananda, A. Advanced Natural Language Processing. In Natural Language Processing Recipes; Apress, 2021; pp. 107–133.
[39] Hasija, Y.; Chakraborty, R. Natural Language Processing. In Hands-On Data Science for Biologists Using Python; CRC Press, 2021; pp. 261–273.
[40] Sri, M. Practical Natural Language Processing with Python; Apress, 2021.
[41] Barella, V.H.; Garcia, L.P.; de Souto, M.C.; Lorena, A.C.; de Carvalho, A.C. Assessing the data complexity of imbalanced datasets. Information Sciences 2021, 553, 83–109.
[42] Azhar, N.A.; Pozi, M.S.M.; Din, A.M.; Jatowt, A. An Investigation of SMOTE based Methods for Imbalanced Datasets with Data Complexity Analysis. IEEE Transactions on Knowledge and Data Engineering 2022, pp. 1–1.
[43] Alruily, M.; .; Shahin, O.R. Sentiment Analysis of Twitter Data for Saudi Universities. International Journal of Machine Learning and Computing 2020, 10, 18–24.
[44] Sun, S.; Luo, C.; Chen, J. A review of natural language processing techniques for opinion mining systems. Information Fusion 2017, 36, 10–25.
[45] Al-Twairesh, N.; Al-Khalifa, H.; Al-Salman, A.; Al-Ohali, Y. AraSenTi-Tweet: A Corpus for Arabic Sentiment Analysis of Saudi Tweets. Procedia Computer Science 2017, 117, 63–72.
[46] Hegazi, M.O.; Al-Dossari, Y.; Al-Yahy, A.; Al-Sumari, A.; Hilal, A. Preprocessing Arabic text on social media. Heliyon 2021, 7, e06191.
[47] Churchill, R.; Singh, L. textPrep: A Text Preprocessing Toolkit for Topic Modeling on Social Media Data. In Proceedings of the Proceedings of the 10th International Conference on Data Science, Technology and Applications. SCITEPRESS - Science and
Technology Publications, 2021.
[48] Thelwall, M.; Buckley, K.; Paltoglou, G.; Cai, D.; Kappas, A. Sentiment strength detection in short informal text. Journal of the American Society for Information Science and Technology 2010, 61, 2544–2558.
[49] Kilimci, Z.H.; Akyokus, S. The Evaluation of Word Embedding Models and Deep Learning Algorithms for Turkish Text Classification. In Proceedings of the 2019 4th International Conference on Computer Science and Engineering (UBMK). IEEE, 2019.
[50] Asudani, D.S.; Nagwani, N.K.; Singh, P. Impact of word embedding models on text analytics in deep learning environment: a review. Artificial Intelligence Review 2023.
[51] Almuzaini, H.A.; Azmi, A.M. Impact of Stemming and Word Embedding on Deep Learning-Based

Arabic Text Categorization. IEEE Access 2020, 8, 127913–127928.

[52] Gamal, D.; Alfonse, M.; El-Horbaty, E.S.M.; Salem, A.B.M. Implementation of Machine Learning Algorithms in Arabic Sentiment Analysis Using N-Gram Features. Procedia Computer Science 2019, 154, 332–340.

[53] Chen, K.; Cosgro, B.; Domfeh, O.; Stern, A.; Korkmaz, G.; Kattampallil, N.A. Leveraging Google BERT to Detect and Measure Innovation Discussed in News Articles. In Proceedings of the 2021 Systems and Information Engineering Design Symposium (SIEDS). IEEE, 2021.

[54] Kamath, U.; Graham, K.L.; Emara, W. Bidirectional Encoder Representations from Transformers (BERT). In Transformers for Machine Learning; Chapman and Hall/CRC, 2022; pp. 43–70.

[55] Sabharwal, N.; Agrawal, A. BERT Model Applications: Question Answering System. In Hands-on Question Answering Systems with BERT; Apress, 2021; pp. 97–137.

[56] Mingua, J.; Padilla, D.; Celino, E.J. Classification of Fire Related Tweets on Twitter Using Bidirectional Encoder Representations from Transformers (BERT). In Proceedings of the 2021 IEEE 13th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM). IEEE, 2021.

[57] Zheng, X.; Zhang, C.; Woodland, P.C. Adapting GPT, GPT-2 and BERT Language Models for Speech Recognition. In Proceedings of the 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE, 2021.

[58] Gaikwad, A.; Rambhia, P.; Pawar, S. An Extensive Analysis Between Different Language Models: GPT-3, BERT and MACAW. Research Square 2022.

[59] Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. CoRR 2018,

[60] González, J.Á.; Hurtado, L.F.; Pla, F. TWilBert: Pre-trained deep bidirectional transformers for Spanish Twitter. Neurocomputing, 2021, 426, 58–69.

[61] Qiang, Y.; Pan, D.; Li, C.; Li, X.; Jang, R.; Zhu, D. Attcat: Explaining transformers via attentive class activation tokens. In Proceedings of the Advances in Neural Information Processing Systems, 2022.

[62] BehnamGhader, P.; Zakerinia, H.; Baghshah, M.S. MG-BERT: Multi-Graph Augmented BERT for Masked Language Modeling. In Proceedings of the Proceedings of the Fifteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-15). Association for Computational Linguistics, 2021.

[63] Chien, C.Y.; Chen, K.Y. A BERT-based Language Modeling Framework. In Proceedings of the Interspeech 2022. ISCA, 2022.

[64] Doreswamy.; Gad, I.; Manjunatha, B.R. Multi-label Classification of Big NCDC Weather Data Using Deep Learning Model. In Soft Computing Systems; Springer Singapore, 2018; pp. 232–241.

[65] El-Shafai, W.; Hemdan, E.E.D. Robust and efficient multi-level security framework for color medical images in telehealthcare services. Journal of Ambient Intelligence and Humanized Computing 2021.

[66] Gad, I.; Elmezain, M.; Alwateer, M.M.; Almaliki, M.; Elmarhomy, G.; Atlam, E. Breast Cancer Diagnosis Using a Machine Learning Model and Swarm Intelligence Approach. In Proceedings of the 2023 1st International Conference on Advanced Innovations in Smart Cities (ICAISC). IEEE, 2023.

[67] Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research 2002, 16, 321–357.

[68] Pratama, F.R.A.; Oktora, S.I. Synthetic Minority Over-sampling Technique (SMOTE) for handling imbalanced data in poverty classification. Statistical Journal of the IAOS 2023, 39, 233–239.