Tech Science Press

check for updates

Multi-Tier Sentiment Analysis of Social Media Text Using Supervised Machine Learning

Hameedur Rahman[1], Junaid Tariq[2,*], M. Ali Masood[1], Ahmad F. Subahi[3], Osamah Ibrahim Khalaf[4] and Youseef Alotaibi[5]

[1]Department of Creative Technologies, Air University, E-9 Islamabad, 44230, Pakistan
[2]Department of Computer Science, National University of Modern Languages, Rawalpindi, Pakistan
[3]Department of Computer Science, University College of Al Jamoum, Umm Al-Qura University, Makkah, 21421, Saudi Arabia
[4]Al-Nahrain Nano-Renewable Energy Research Center, Al-Nahrain University, Baghdad, 10072, Iraq
[5]Department of Computer Science, College of Computer and Information Systems, Umm Al-Qura University, Makkah, 21955, Saudi Arabia
*Corresponding Author: Junaid Tariq. Email: junaid.tariq@numl.edu.pk

Abstract: Sentiment Analysis (SA) is often referred to as opinion mining. It is defined as the extraction, identification, or characterization of the sentiment from text. Generally, the sentiment of a textual document is classified into binary classes i.e., positive and negative. However, fine-grained classification provides a better insight into the sentiments. The downside is that fine-grained classification is more challenging as compared to binary. On the contrary, performance deteriorates significantly in the case of multi-class classification. In this study, pre-processing techniques and machine learning models for the multi-class classification of sentiments were explored. To augment the performance, a multi-layer classification model has been proposed. Owing to similitude with social media text, the movie reviews dataset has been used for the implementation. Supervised machine learning models namely Decision Tree, Support Vector Machine, and Naïve Bayes models have been implemented for the task of sentiment classification. We have compared the models of single-layer architecture with multi-tier model. The results of Multi-tier model have slight improvement over the single-layer architecture. Moreover, multi-tier models have better recall which allow our proposed model to learn more context. We have discussed certain shortcomings of the model that will help researchers to design multi-tier models with more contextual information.

1 Introduction

In the last few decades, the number of users on social media has grown exponentially. Facebook has 1.28 billion active users and Twitter has 241 million active users [1]. These platforms serve as the hub for people to express their opinions. Thus, a massive amount of data is provided by Social Media platforms, which indicate the behavior and sentiment of the users. Sentiment Analysis (SA) of this data can be beneficial in many domains. SA has been a focus of many researchers owing to its vast applications in the industry [2]. It brings forth an automated way to mine the opinions and sentiments of the target audience. Organizations and companies can gain financial benefits by analyzing the sentiments of consumers. The government can understand the public's perceptions and sentiments regarding their policies [3].

While the common supervised ML models namely Decision Tree, Support Vector Machine (SVM), and Naïve Bayes ML algorithms have been implemented widely for different tasks in multiple domains, such as data privacy, and healthcare security assessment [4,5].

The most naïve approach to SA is to classify the text into binary classes i.e., positive and negative. Binary ML classifiers yield high accuracy [6]. However, multi-class classification proves to be more beneficial as it provides an in-depth insight into the sentiments. Thus, the decision-makers are better equipped to make accurate decisions. Aspect-based SA is another tool of paramount importance that gives the awareness of the users' sentiment regarding a specific aspect of a particular entity. The problem arises when baseline ML algorithms fail to give an optimal accuracy for multi-class classification problems. Naïve Bayes and SVM are the most commonly used algorithms for SA [7,8]. These algorithms cease to perform optimally in a social media text. Degradation in performance arises because social media text frequently contains erroneous spellings, domain-centric slang, spam reviews, and negation in sentences. Due to these challenges, the SA of social media becomes a challenging task. Many researchers have put in efforts to enhance the performance of the multi-class classifier. To overcome this challenge, various architectures have been proposed. However, on account of the complexity of the social media text, the data embodies high-level features and is not linearly separable. Thus, the baseline ML models are incapable of completely capturing and learning these features. However, more complex deep learning (DL) models such as Recurrent Neural Network (RNN), Convolutional Neural Network (CNN) and Bidirectional Encoder Representation of Transformer (BERT) can perform more optimally [9,10].

Besides the model selection and designing an optimal architecture, pre-processing of the textual data plays a key role in augmenting the overall performance [11]. Thus, effective preprocessing of the social media text has been a focus of many researchers. The baseline pre-processing of textual data involves a number of crucial steps: 1) stop words elimination, 2) tokenization 3) word normalization, and 4) vectorization [8]. Which language model i.e., unigram, bigram, or trigram fits best the particular NLP task at hand and comes under the umbrella of pre-processing techniques.

In this study, a multi-tier model has been proposed to augment the performance of multiclass sentiment classification. The proposed model has been implemented on the Sentiment Analysis on Movie Reviews data set [12]. Movie Reviews are classified into 5 classes i.e., positive, strongly positive, negative, strong negative, and neutral. Before the training and evaluation of the model, pre-processing techniques have been implemented for the SA task. For the implementation pre-processing NLP library, NLTK (natural language toolkit) has been utilized. TreeBank word tokenizer tokenizes the text data. WordNet Lemmatizer fulfills the need for text normalization. For the removal of stop words, the built-in list of stop words has been used. Lastly, a term frequency–Inverse document frequency (TF-IDF) vectorization is performed to represent the textual data into numerical vector space that can be

used to train the ML model. The multi-tier model decomposes the multiclass classification into binary and ternary classification sub-tasks. The ML models perform more optimally in case of binary and ternary classification. However, the challenges such as negation, spam reviews, unbalanced data over classes, and spelling errors still contribute to the degradation of the accuracy. The performance metrics of the multi-tier model are compared with those of the single-tier model. Also, the performance metrics of the binary and ternary classifiers that constitute the multi-tier model are analyzed. It can be noted that as the number of classes is added the performance is significantly degraded. The aim of this study is to identify whether a multi-tier architecture is useful for sentiment classification, if yes, then we can use it for other sentiment classification task. Otherwise, we will identify the short coming and possible solutions. This will address the following research questions:

1. Will the multi-tier model gain more useful information than single-tier model?
2. What are the short comings of multi-tier model?
3. What are the possible solutions to improve the performance of multi-tier model?

The rest of the paper constitutes the following sections. Section 3 presents the related work done in the area of SA. In Section 4 proposed methodology is discussed. Results are displayed in Section 5. Section 6 results are discussed and finally, Section 7 concludes the paper.

2 Related Work

The computational study of people's opinions, sentiments, emotions, and attitudes towards entities such as products, services, issues, events, topics, and their attributes is known as SA [2,3]. SA is divided into five sub-tasks: subjectivity classification, sentiment classification, opinion spam detection, implicit language detection, and aspect extraction [8]. All these tasks rely on sentence or document level, which is later applied to machine learning, deep learning, lexicon-based, or a hybrid approach for classification. The sentiment classification model needs to use certain features. Therefore, Chan et al. [6] proposed six categories of features, i.e., Terms and Frequency, Parts of Speech, Rules and Opinions, Sentiment Shifters, and Syntactic Dependency.

2.1 Feature Representation of Text

The most common representation for features in sentiment classification is Bag of Words (BOW), which represents a document in terms of the presence or absence of words [7]. However, the presence or absence cannot represent the value/quality of the words. For this sake, the TF-IDF scheme is used to give more weight to the important words in the corpus. The drawback of this approach is that size of the sparse feature vector increases with the increase in the document size. To overcome this limitation, one approach is to optimize the feature vector with the use of lexicon and machine learning methods [13]. Moreover, a genetic algorithm (GA) was implemented for feature reduction. The proposed pipeline is comprised of data cleaning, preprocessing, and an analysis engine. The fitness function used is based on the lexical database (SentiWordNet), which calculates the distance in terms of polarity score.

The lexicon and TF-IDF may not accurately take the semantics of the text. To incorporate the semantics researchers have used word embedding. Word embedding converts the words of similar context into similar dense vectors in multi-dimensional space, each dimension represents some aspect of the word. The authors in [14] used the pre-trained word embedding e.g., Word2Vec and GloVe to identify the sentiment. However, using pre-trained word embedding has a few drawbacks: firstly, a large corpus is required for an optimum representation of word vectors. Another problem arises because of the inability to consider the context of the document e.g., "beetle" can be the name of a car or insect

but both have the same representation [14]. To address these issues, researchers have concatenated Word2Vec, Pos2Vec, Word-position2Vec (Wp2V), and Lexicon2Vec (L2V) for sentiment classification [15]. Pos2Vec is a Part of Speech (POS) tag vector providing syntactic information. Wp2V gives the location of a word concerning the ordering of the sentence. L2V is the sentiment score provided by the Lexicon. Experimental results showed performance improvement.

Apart from this, [15] suggested the domain-specific word vector performs better than the pre-trained vector. The authors researched Ebola and Zika virus outbreaks. Initially, when a certain event occurs, the Twitter corpus is scanty to train a vector. The solution to this problem was to use prior scholarly text to train the unstructured twitter corpus.

Although the ML models with complex word embedding have shown good performance, they are limited in retaining contextual information of the long text. Target Dependent BERT [10] for sentiment classification while considering the target.

2.2 Multiclass Classification

Generally, sentiment classification relies on three classes, i.e., negative, neutral, and positive. However, these classes may not depict the real-world scenario. Therefore, Multi-Tier model [6,16] architecture base models were proposed. The authors in [6] proposed classification was an efficient way, for this sake, they used spark. In their approach, they trained three models to identify the sentiment of the tweets. The result showed that Linear Support Vector Machine (SVM) generated better accuracy, but it took more time to classify unseen data. Similarly, [17] used a multi-class classification of tweets. However, the authors proposed to consider the sentiment classification as an ordinal regression problem where labels are considered as ordinal data. In this approach, a scoring and balancing methodology were applied for the classification of tweets. This approach presented considerably good results. However, some data were misclassified due to the removal of emoticons that depicted relevant sentiment. In general, replacing emoticons with sentiment can mislead the classifier.

The ML-based approach requires hand-crafted feature engineering. Hand-crated features may induce a bias in the model. Therefore, researchers are more driven toward deep learning methods. Reference [3] conducted a comparative analysis of Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) on pre-trained word embedding. Based on their findings, the authors concluded that the combination of CNN and RNN results in better performance. It is important to note that CNN cannot hold long sequence semantics in the text. For this sake, Dilated CNN was proposed, which has three parallel dilated CNN layers. The idea of the parallel layer is to reduce dimension and increase accuracy without losing information. Similarly, CNN and RNN together were used by [11] to make use of long-term dependencies in classification. In their approach, CNN is applied to extract high-level features, which are then applied to the attention layer to calculate the attention score. Finally, these features and attention scores were applied on the RNN to learn long-term dependencies in the classification. Experimental results on standard datasets show performance improvement.

The sentiment classification is highly domain-specific i.e., the polarity can alter in different domains. Therefore, proposed Domain-specific sentiment analysis (DSSA-H) [18] was proposed by. Domain-specific tweets were retrieved using Random Forest. Domain Adaptive Neural Network (DANN) was applied for sentiment classification. DANN embeds a domain-adversarial component, which relies less on the training data. The proposed method can be applied to other domains, making it more scalable.

2.3 Negation Detection

In sentiment classification, negation plays a key role as it may alter the ongoing sentiments represented by the text. Thus, negation detection needs to be focused upon. Otherwise, the classification may result in incorrect sentiment. Saeed et al. [19] enhanced the performance of sentiment classification by negation handling, emoticons detection, and removal of spam reviews. The authors used the apriori algorithm to extract the features from the reviews. Scores were assigned to the emoticon using the emoticon dictionary. Sentiments of the opinion words and emoticons were calculated. However, the scores were re-evaluated based on the presence of the word in the negation list. Then data were classified into spam or truthful with the help of processing of content properties, part of speech (POS) tags, sentiment score, and enhanced bag of words (BOW). Spam data was eliminated and SA was applied only to truthful reviews. The results show a significant improvement as compared to the state-of-the-art techniques.

Negation detection is crucial for accurately classifying the sentiment [13]. Negation detection tasks can be performed using two approaches i.e., Rule-based and ML-based. In real-life scenarios, both methods may suffer to some extent. To address this issue, an optimal solution was proposed by the authors based on Reinforcement Learning (RL). The RL is based on a text processing function that evaluates the correlation between the document text and the gold standard assigned to it. Then the tone of the document is calculated to negate or inverse negate a word. Evaluation of the model shows promising results. The proposed methodology removed the bias caused by human error.

2.4 Aspect Based SA

With the advent of deep learning, most approaches have used deep learning methods for sentiment classification. Forty deep learning models were analyzed by [20], comparing performance metrics of aspect term detection, aspect category detection, and sentiment polarity. The author analyzed that CNN, Long-Short Term Memory (LSTM), Gated Recurrent Unit (GRU), and their variants are most frequently used.

The deep-learning models learn the parameters and features automatically. The learned features may give more importance to certain aspects than others. However, these aspects in some cases may not relate to the real world. To model practical scenarios, it is integral to explicitly give importance to certain aspects. This can be achieved by utilizing an attention mechanism. Coattention-MemNet and Coattention-LSTM network based on a co-attention mechanism was proposed by [21]. The proposed methodology focused on target and context level attention rather than assigning an average score. The context level attention to target words has shown promising results.

Deep learning models like RNN, LSTM, and BERT have proven efficient in capturing the semantic relation between target and context words. The deriving of contextual and semantic relationships may require more computational power. Therefore, [22] proposed a lightweight Graph Convolutional Neural Network (GCNN) based DNet (Distillation Network). This model used fewer resources with a minute reduction in performance metrics. In comparison with BERT, DNet decreases the model size fifty times and increases the responsiveness by twenty-four times.

A lexicon-based approach was incorporated by [23] that was proven to be more context-oriented. For the development of Aspect Based Sentiment Analysis ABSA, the existing Static lexicon was combined to improve accuracy. The proposed Aspect-Based Frequency Based Sentiment Analysis (ABFBSA) was proposed in which the token distance was used to evaluate the label of the nearest aspect to a given word. A list of seven negation words was used to deal with the negation. ABSA

is a three-step process i.e., identification, classification, and aggregation [24]. There are Frequency-based and Syntax-based methods for aspect detection. Considering each method had its drawbacks, hybrid models were graded highly. For SA, one approach was to break text into parts, evaluate their sentiments then syntactically merge. However, aspect detection and SA can be jointly performed [25–31]. The future of ABSA is inclined toward more semantically rich and concept-centric rather than word-based.

While performing ABSA, the biggest challenge is of annotating the data with sentiment and aspect tags. It is a time-consuming and expensive task since erroneous annotation can induce bias. W2VLDA was implemented by [26] to classify aspects into a category, separate aspect-term, opinion-word, and SA for any given domain and language. The proposed methodology required a single seed word for each domain aspect and a single positive and negative word independent of the domain [32–39]. This made it minimally supervised. The tasks of aspect detection and SA were jointly performed. Varying the seed words of each aspect does not affect the accuracy of the sentiment [40–46].

3 Proposed Model and Implementation

3.1 Multi-Tier Architecture

Multi-tier architecture has been proposed for the multi-class SA. Their Multi-tier model hierarchically implements ternary and binary classifiers. Hence Machine Learning models are trained. Fig. 1 shows the flow diagram of the proposed methodology. The multi-tier model is comprised of three models. The training of the models takes simultaneously. Training data for M1 consists of positive, negative, and neutral text. Thus, M1 is trained to classify text into ternary classes. M2 and M3 are trained for binary classification. M2 is trained for positive and strong positive data. M3 is trained for negative and string negative text. Each classifier was trained using Naïve Bayes, Decision Tree, and SVM [47–53].

3.2 Preprocessing Module

Effective preprocessing is the foremost part of any ML algorithm. Since the ML algorithm does not work for textual data, the preprocessing needs to be applied to convert textual data into numerical vectors. The preprocessing is carried out with the help of the NLTK library in python [54–62]. The preprocessing module as shown in Fig. 2 works according to the following steps

1. A list of English stop words present in the NLTK corpus was used to remove stop words.
2. Punctuations and special characters were removed.
3. After the data was cleaned, Lemmatization was applied. This normalizes the words into their root words according to the dictionary.
4. TF-IDF vectorizer was used to convert the textual data into vectors. TF-IDF is the product of Term Frequency and Inverse document frequency.

$$\text{TF-IDF}\,(t, d = tf\,(t, d) * IDF\,(t) \tag{1}$$

$$IDF\,(t) = \log\left[(1 + n)/(1 + df\,(t))\right] + 1 \tag{2}$$

Eqs. (1) and (2) show how TF-IDF is calculated, n is the total number of documents, df(t) is the document frequency of t; the document frequency is the number of documents in the document set that contain the term t [11]. The idea behind TF-IDF is that the frequently occurring words do not

hold information regarding the sentiments. Thus, their contribution is scaled down and more weight is given to words that occur less frequently. Preprocessing was applied to the entire dataset and thus the feature vector caters to the words that are present in both test and training data. Thus, the preprocessed data is split into dev, train, and test sets according to the 10:70:20 ratio, respectively.
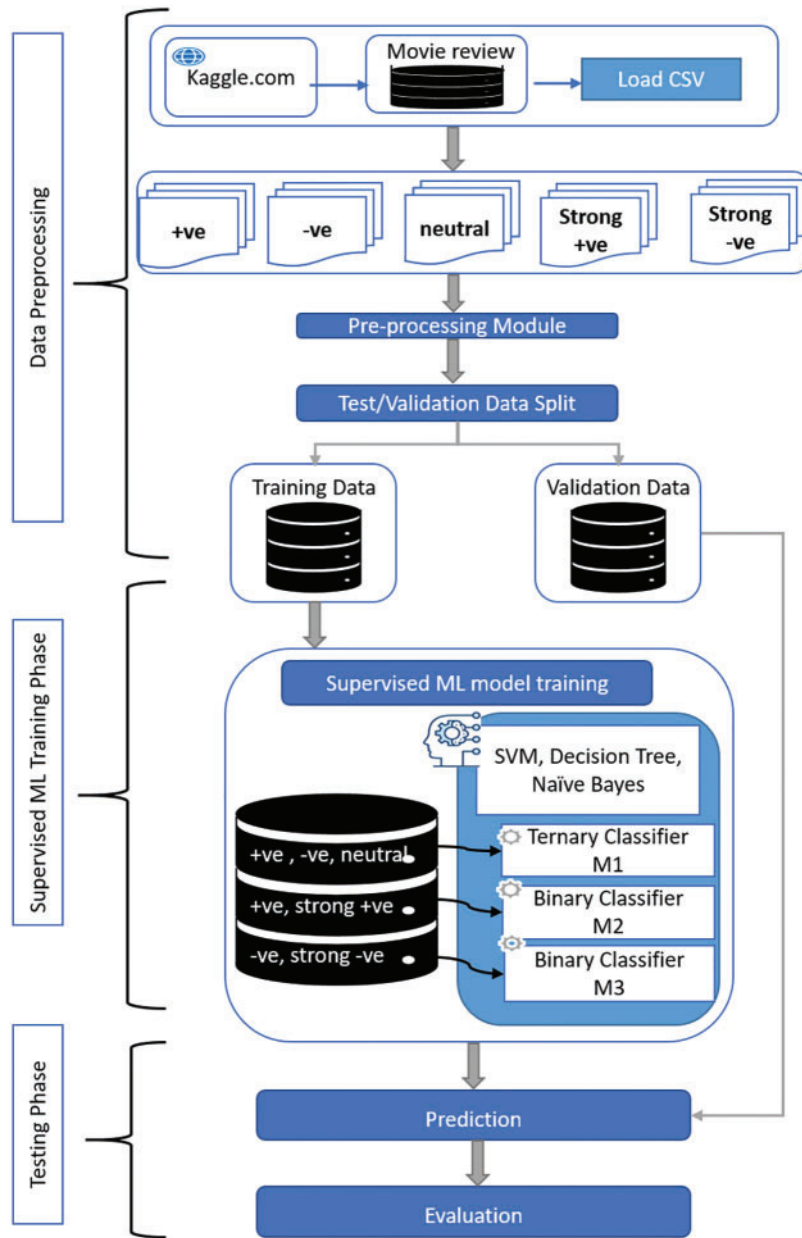

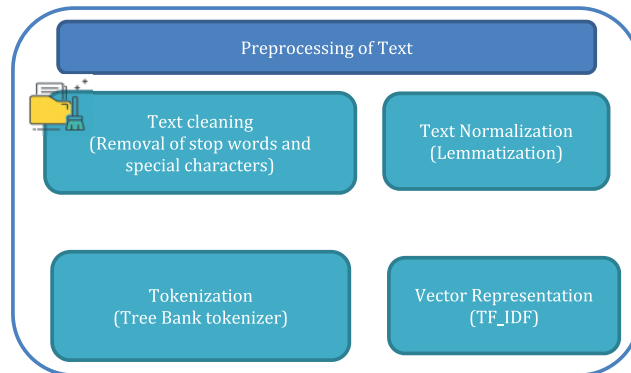**Figure 1:** End-to-End workflow of the multi-tier architecture

Figure 2: Steps for pre-processing of textual data

3.3 Experimental Setup

3.3.1 Dataset

The performance of the ML model is highly data-driven. A labelled dataset is required to perform supervised ML. Assigning 5 levels of sentiment to social media is a complex task. Labelling can either be done through human annotation or ML. Human annotation, although good in accuracy, is laborious. Thus, annotated Sentiment Analysis on Movie Reviews data set [12] tends to provide an optimal solution to this problem. In this study, data implementation has been performed on a publicly available movie review dataset. Table 1 shows the distribution of movie reviews among the classes. Most of the reviews belong to the neutral class. The extreme classes i.e., strong positive and string neutral contain minimum classes.

Table 1: Distribution of labels over entire data

Sentiment	label	Number of reviews
Strong positive	0	7072
Positive	1	27273
Neutral	2	79582
Negative	3	32927
Strong neutral	4	9206

The data is not equally distributed among the labels. However, it is expected to have majority reviews to have neutral sentiments. The issue arises while training the Model with imbalanced data. The ML model tends to be biased towards the majority class. The preprocessed data is split into dev, train, and test sets according to the 10:70:20 ratio, respectively. Fig. 3 shows the histogram of the data distribution is created using the labels given in the dataset. As seen in Fig. 3, the data has more neutral sentiments, i.e., "2", which almost three times more than positive or negative category. Therefore, it brings imbalance in the dataset, which results in model predicting more neutral categories than others.

Fig. 4 displays the distribution of the number of words. Most of the phrases comprise 10–20 words. Fig. 5 shows how the sentiments are distributed among reviews comprised of long phrases. Fig. 6 shows the 15 most frequent words. These frequent words are known as stop words and are part of the structure of the sentence.
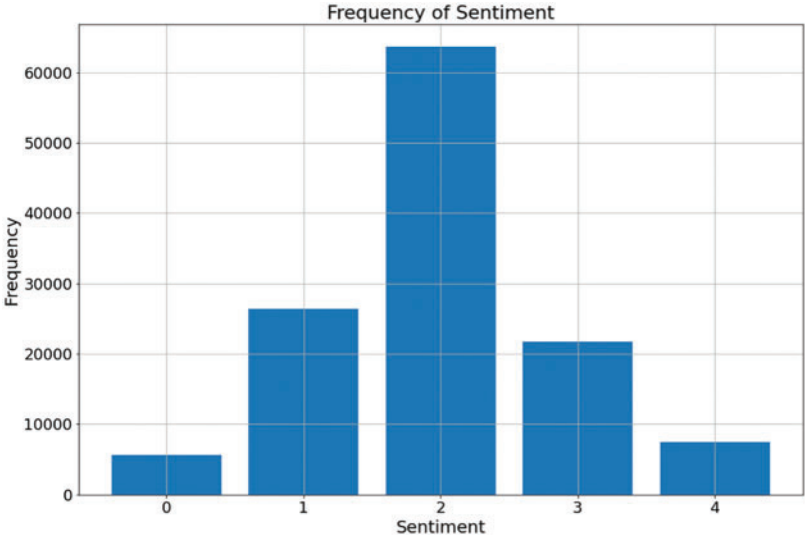

Figure 3: Histogram of distribution of sentiments from the dataset. Neutral sentiment is more dominant in the dataset
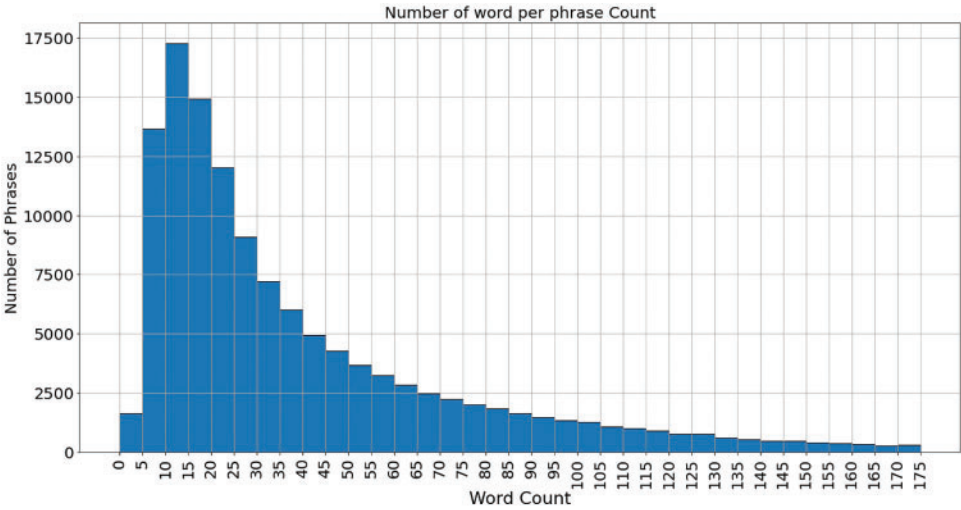

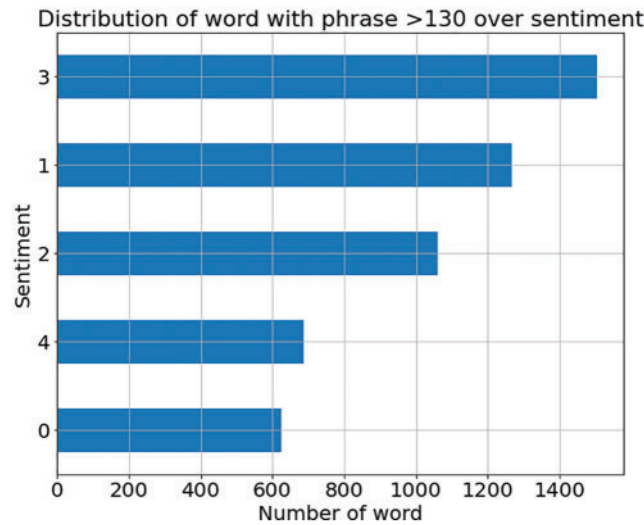**Figure 4:** Number of words per phrase count

Figure 5: Distribution of phrases with word count >170 concerning sentiment
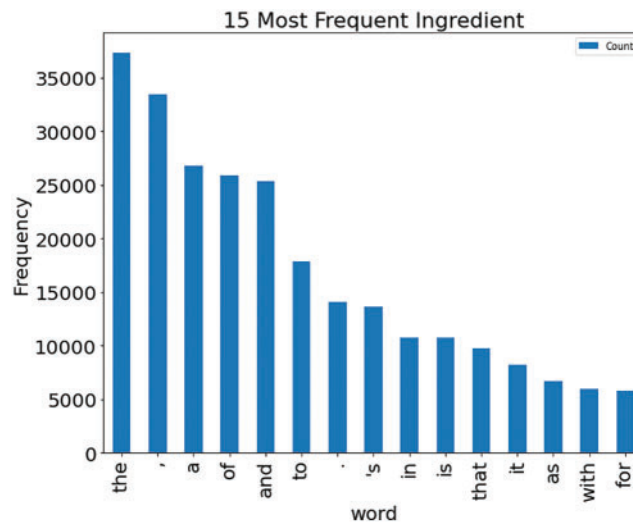

Figure 6: 15 most frequent words

Word cloud, pictorially represents the important words in the corpus. The size of a word depicts how frequently it appears in the corpus. Fig. 7 shows the words cloud of the dataset.

3.3.2 Implementation of Multi-Tier Model

Once the models are trained, sentiments of text data can be referenced. The prediction phase involves predicting the label of unseen movie reviews. The prediction takes place in two steps. First, the ternary model classifies the data into positive, negative, and neutral. In the second step, the positive predicted data are fed into M2 that further categorized into strong positive and positive. The negative predicted data is fed into the M3 that further categorized into strings negative and negative. The workflow of the prediction phase is shown in Fig. 8. Model performance was quantified by evaluating

precision, recall, F1-score, and accuracy. The implementation was performed with the help of the Scikit-Learn library [63–66].

Figure 7: Pictorial representation of most frequent words. (Word cloud)

Figure 8: Prediction phase of a multi-tier model

4 Results

Three classifiers have been implemented. Performance metrics of multi-tier, single-tier and individual classifiers have been evaluated. Performance of individual Model is improved. Table 2 shows the comparison of results of the multi-tier and single-tier models. Table 3 depicts the results of ternary classifier M1. Tables 4 and 5 tabulate the results of binary classifiers M2 and M3, respectively.

Table 2: Performance metrics of the multi-tier and single-tier model

Classifier	Class	Support	Multi-tier model				Single-tier model			
			Precision	Recall	F1 score	Accuracy	Precision	Recall	F1 score	Accuracy
SVM	0	2186	0.22	0.18	0.15	0.52	0.21	0.13	0.16	0.51
	1	8777	0.42	0.31	0.24		0.38	0.21	0.27	
	2	23109	0.59	0.67	0.79		0.58	0.80	0.67	
	3	9853	0.40	0.33	0.29		0.40	0.27	0.33	
	4	2892	0.30	0.26	0.23		0.29	0.23	0.25	
Decision tree	0	2186	0.23	0.18	0.15	0.52	0.22	0.15	0.18	0.51
	1	8777	0.35	0.33	0.31		0.35	0.29	0.32	
	2	23109	0.63	0.68	0.75		0.62	0.76	0.68	
	3	9853	0.40	0.37	0.35		0.39	0.31	0.35	
	4	2892	0.34	0.22	0.16		0.26	0.16	0.20	
Naïve Bayes	0	2186	0.38	0.04	0.02	0.56	0.37	0.02	0.03	0.55
	1	8777	0.47	0.31	0.23		0.47	0.18	0.26	
	2	23109	0.59	0.71	0.88		0.57	0.91	0.71	
	3	9853	0.48	0.42	0.38		0.48	0.30	0.37	
	4	2892	0.41	0.08	0.04		0.31	0.04	0.08	

Table 3: Performance metrics of ternary model M1

Classifier	Class	Precision	Recall	F1 score	Support	Accuracy
SVM	1	0.71	0.62	0.66	10396	0.73
	2	0.72	0.80	0.76	23816	
	3	0.74	0.69	0.71	12606	
Decision tree	1	0.59	0.59	0.59	10383	0.67
	2	0.70	0.75	0.73	23864	
	3	0.67	0.58	0.62	12571	
Naïve Bayes	1	0.72	0.47	0.57	10383	0.69
	2	0.66	0.84	0.74	23864	
	3	0.75	0.58	0.66	12571	

Table 4: Performance metrics of binary model M2

Classifier	Class	Precision	Recall	F1 score	Support	Accuracy
SVM	0	0.50	0.36	0.42	2111	0.80
	1	0.85	0.91	0.88	8193	
Decision tree	0	0.42	0.39	0.40	2111	0.76
	1	0.85	0.86	0.85	8193	
Naïve Bayes	0	0.58	0.21	0.31	2111	0.81
	1	0.83	0.96	0.89	8193	

Table 5: Performance metrics of binary model M3

Classifier	Class	Precision	Recall	F1 score	Support	Accuracy
SVM	3	0.84	0.90	0.87	9847	0.79
	4	0.52	0.38	0.44	2793	
Decision tree	3	0.83	0.88	0.85	9847	0.76
	4	0.45	0.35	0.40	2793	
Naïve Bayes	3	0.81	0.96	0.88	9847	0.79
	4	0.60	0.20	0.30	2793	

The multi-tier architecture does not ameliorate the performance significantly. The possible reasons for the ineffectualness of the proposed model are discussed in the next section.

5 Discussion

The multi-tier model results are unremunerative. There is some improvement in accuracy with the base models. A few reasons for the ineffectiveness of the proposed model are put forth. First, the main drawback that leads to deterioration of the performance is that as the data increases the feature vector size increases manifold. Thus, it is quite likely that the ML model may not perform well on sparse matrix. One possible solution is to apply the Principal Component Analysis (PCA) algorithm to reduce the number of features. Second, the data is highly imbalanced. It is expected that humans who write reviews are mainly of neutral polarity. However, the imbalance of classes can cause a bias in the ML model, which alters accuracy. The individual models do not have a huge class imbalance in the training data. Third, the ML models are unable to learn the high-level features of the textual data. For example, in this approach the ML model uses TF-IDF score, which scores the importance of the word not the semantics and relationship of the word with subject and object. Therefore, the ML model may not identify the semantics or if user writes a comment in a different way. So, it is important to learn the structure/relationship among words, thus the model does not only rely on wording rather than other high level features. For this sake, word embedding along with Deep Learning (DL) can be useful. Fourth, the textual data might contain sarcasm and negation. This alters the polarity of the text. Thus, negation handling and spam detection are desiderata. Finally, Sentiments are written by humans so whenever a human writes a review he/she links the relationship between subject and object.

For this sake, a graph based structure to model the reviews of the user can be useful to identify the perspective or context of a user.

As shown in Table 2, the recall of the multi-tier model is mostly better than the recall of single-tier architecture. This shows that the proposed model is able to learn more overall context.

The proposed model does not extract the semantic and syntactic properties of the text. To incorporate the syntax, word embedding can be applied. Various DL methods can learn the long-term properties of the text data. To keep the semantics of the text, Glove can be applied.

6 Conclusion

Fine-grained SA has been an area of interest for researchers. However, more efforts are being put into making SA more context-oriented and aspect-based. In this study, a multi-tier architecture has been proposed for the multi-class classification of sentiments of the text. The classifiers were trained using three supervised ML algorithms i.e., Naïve Bayes, Support Vector Machine and Decision Tree. A comparison of the three algorithms showed that Naïve Bayes presented the best accuracy. In future work, we aim to improve the performance by incorporating Negation handling and spam review. Feature optimization increases the efficiency of the model. Emoticons and POS tagging can be concatenated with the feature vector to deliver true insight into the sentiment. Hence, these features can be made part of the preprocessing module to yield promising results in the future.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] N. Petrocchi, A. Asnaani, A. P. Martinez, A. Nadkarni and S. G. Hofmann, "Differences between people who use only Facebook and those who use Facebook plus Twitter," *International Journal of Human-Computer Interaction*, vol. 31, no. 2, pp. 157–165, 2015.

[2] A. Ligthart, C. Catal and B. Tekinerdogan, "Systematic reviews in sentiment analysis: A tertiary study," *Artificial Intelligence Review*, vol. 54, no. 7, pp. 4997–5053, 2021.

[3] L. Yue, W. Chen, X. Li, W. Zuo and M. Yin, "A survey of sentiment analysis in social media," *Knowledge and Information Systems*, vol. 60, no. 2, pp. 617–663, 2019.

[4] M. Ahmad, J. F. Al-Amri, A. F. Subahi, S. S. Khatri, A. H. Nadeem *et al.,* "Healthcare device security assessment through computational methodology," *Computer Systems Science and Engineering*, vol. 41, no. 2, pp. 811–828, 2022.

[5] A. H. Seh, J. F. Al-Amri, A. F. Subahi, A. Agrawal, R. Kumar *et al.,* "Machine learning based framework for maintaining privacy of healthcare data," *Intelligent Automation & Soft Computing*, vol. 29, pp. 697–712, 2021.

[6] W. N. Chan and T. Thein, "A comparative study of machine learning techniques for real-time multi-tier sentiment analysis," in *1st IEEE Int. Conf. on Knowledge Innovation and Invention (ICKII)*, Jeju Island, Korea (South), pp. 90–93, 2018.

[7] F. Iqbal, J. M. Hashmi, B. C. M. Fung, R. Batool, A. M. Khattak *et al.,* "A hybrid framework for sentiment analysis using genetic algorithm based feature reduction." *IEEE Access*, vol. 7, pp. 14637–14652, 2019.

[8]   M. Alam, F. Abid, C. Guangpei and L. V. Yunrong, "Social media sentiment analysis through parallel dilated convolutional neural network for smart city applications," *Computer Communications*, vol. 154, pp. 129–137, 2020.

[9]   M. Usama, B. Ahmed, E. Song, M. S. Hossain, M. Alrashoud *et al.,* "Attention-based sentiment analysis using convolutional and recurrent neural network," *Future Generation Computer Systems*, vol. 113, pp. 571–578, 2020.

[10] G. Zhengjie, A. Feng, X. Song and X. Wi, "Target-dependent sentiment classification with BERT," *IEEE Access*, vol. 7, pp. 154290–154299, 2019.

[11] A. P. Pimpalkar and R. J. R. Raj, "Influence of pre-processing strategies on the performance of ML classifiers exploiting TF-IDF and BOW features," *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, vol. 9, no. 2, pp. 49–68, 2020.

[12] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning *et al.,* "Recursive deep models for semantic compositionality over a sentiment treebank," in *Conf. on Empirical Methods in Natural Language Processing*, Stanford, CA 94305, USA, pp. 1631–1642, 2013.

[13] N. Pröllochs, S. Feuerriegel, B. Lutz and D. Neumann, "Negation scope detection for sentiment analysis: A reinforcement learning framework for replicating human interpretations," *Information Sciences*, vol. 536, pp. 205–221, 2020.

[14] S. M. Rezaeinia, R. Rahmani, A. Ghodsi and H. Veisi, "Sentiment analysis based on improved pre-trained word embeddings," *Expert Systems with Applications*, vol. 117, pp. 139–147, 2019.

[15] A. Khatua, A. Khatua and E. Cambria, "A tale of two epidemics: Contextual Word2Vec for classifying Twitter streams during outbreaks," *Information Processing & Management*, vol. 56, no. 1, pp. 247–257, 2019.

[16] W. N. Chan and T. Thein, "Multi-tier sentiment analysis system with sarcasm detection: A big data approach," (Doctoral dissertation, *MERAL Portal*), 2018.

[17] E. S. Saad and J. Yang, "Twitter sentiment analysis based on ordinal regression," *IEEE Access*, vol. 7, pp. 163677–163685, 2019.

[18] F. Yao and Y. Wang, "Domain-specific sentiment analysis for tweets during hurricanes (DSSA-H): A domain-adversarial neural-network-based approach," *Computers, Environment and Urban Systems*, vol. 83, pp. 101522, 2020.

[19] N. M. K. Saeed, N. A. Helal, N. L. Badr and T. F. Gharib, "An enhanced feature-based sentiment analysis approach," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 2, pp. e1347, 2020.

[20] H. H. Do, P. W. C. Parsad, A. Maag and A. Alsadoon, "Deep learning for aspect-based sentiment analysis: A comparative review," *Expert Systems with Applications*, vol. 118, pp. 272–299, 2019.

[21] C. Yang, H. Zhang, B. Jiang and K. Li, "Aspect-based sentiment analysis with alternating coattention networks," *Information Processing & Management*, vol. 56, no. 3, pp. 463–478, 2019.

[22] F. Ren, L. Feng, D. Xiao, M. Cai and S. Cheng, "DNet: A lightweight and efficient model for aspect based sentiment analysis," *Expert Systems with Applications*, vol. 151, pp. 113393, 2020.

[23] M. E. Mowlaei, M. S. Abadeh and H. Keshavarz, "Aspect-based sentiment analysis using adaptive aspect-based lexicons," *Expert Systems with Applications*, vol. 148, pp. 113234, 2020.

[24] K. Schouten and F. Frasincar, "Survey on aspect-level sentiment analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 3, pp. 813–830, 2015.

[25] A. G-Pablos, M. Cuadros and G. Rigau, "W2VLDA: Almost unsupervised system for aspect based sentiment analysis," *Expert Systems with Applications*, vol. 91, pp. 127–137, 2018.

[26] Z. Drus and H. Khalid, "Sentiment analysis in social media and its application: Systematic literature review," *Procedia Computer Science*, vol. 161, pp. 707–714, 2019.

[27] A. Koumpouri, I. Mporas and V. Megalooikonomou, "Evaluation of four approaches for sentiment analysis on movie reviews the kaggle competition," in *Proc. of the 16th Int. Conf. on Engineering Applications of Neural Networks (INNS)*, Rhodes Island, Greece, pp. 1–5, 2015.

[28] N. Jha, D. Prashar, O. I. Khalaf, Y. Alotaibi, A. Alsufyani *et al.,* "Blockchain based crop insurance: A decentralized insurance system for modernization of Indian farmers," *Sustainability*, vol. 13, no. 16, pp. 8921, 2021.

[29] A. Sundas, S. Badotra, Y. Alotaibi, S. Alghamdi and O. I. Khalaf, "Modified bat algorithm for optimal vm's in cloud computing," *Computers, Materials & Continua*, vol. 72, no. 2, pp. 2877–2894, 2022.

[30] Y. Alotaibi, "A new meta-heuristics data clustering algorithm based on tabu search and adaptive search memory," *Symmetry*, vol. 14, no. 3, pp. 623, 2022.

[31] D. Anuradha, N. Subramani, O. Khalaf, Y. Alotaibi, S. Alghamdi *et al.,* "Chaotic search-and-rescue-optimization-based multi-hop data transmission protocol for underwater wireless sensor networks," *Sensors*, vol. 22, no. 2867, 2022.

[32] S. S. Rawat, S. Alghamdi, G. Kumar, Y. Alotaibi, O. I. Khalaf *et al.,* "Infrared small target detection based on partial sum minimization and total variation," *Mathematics*, vol. 10, pp. 671, 2022.

[33] P. Mohan, N. Subramani, Y. Alotaibi, S. Alghamdi, O. I. Khalaf *et al.,* "Improved metaheuristics-based clustering with multihop routing protocol for underwater wireless sensor networks," *Sensors*, vol. 22, no. 4, pp. 1618, 2022.

[34] Y. Alotaibi, "A new database intrusion detection approach based on hybrid meta-heuristics," *CMC-Computers, Materials & Continua*, vol. 66, no. 2, pp. 1879–1895, 2021.

[35] P. Jayapradha, Y. Alotaibi, O. I. Khalaf and S. Alghamdi, "Heap bucketization anonymity-an efficient privacy-preserving data publishing model for multiple sensitive attributes," *IEEE Access*, vol. 10, pp. 28773–28791, 2022.

[36] K. Lakshmanna, N. Subramani, Y. Alotaibi, S. Alghamdi, O. I. Khalaf *et al.,* "Improved metaheuristic-driven energy-aware cluster-based routing scheme for IoT-assisted wireless sensor networks," *Sustainability*, vol. 14, no. 13, pp. 7712, 2022.

[37] Y. Alotaibi, "Automated business process modelling for analyzing sustainable system requirements engineering," in *2020 6th IEEE Int. Conf. on Information Management (ICIM)*, London, United Kingdom, pp. 157–161, 2020.

[38] Y. Alotaibi and A. F. Subahi, "New goal-oriented requirements extraction framework for e-health services: A case study of diagnostic testing during the COVID-19 outbreak," *Business Process Management Journal*, vol. 28, no. 1, pp. 273–292, 2021.

[39] N. Subramani, P. Mohan, Y. Alotaibi, S. Alghamdi and O. I. Khalaf, "An efficient Metaheuristic-based clustering with routing protocol for underwater wireless sensor networks," *Sensors*, vol. 22, pp. 415, 2022.

[40] S. Rajendran, O. I. Khalaf, Y. Alotaibi and S. Alghamdi, "MapReduce-Based big data classification model using feature subset selection and hyperparameter tuned deep belief network," *Scientific Reports*, vol. 11, no. 1, pp. 1–10, 2021.

[41] R. Rout, P. Parida, Y. Alotaibi, S. Alghamdi and O. I. Khalaf, "Skin lesion extraction using multiscale morphological local variance reconstruction based watershed transform and fast fuzzy c-means clustering," *Symmetry*, vol. 13, no. 11, pp. 2085, 2021.

[42] S. Bharany, S. Sharma, S. Badotra, O. I. Khalaf, Y. Alotaibi *et al.,* "Energy-efficient clustering scheme for flying ad-hoc networks using an optimized LEACH protocol," *Energies*, vol. 14, no. 19, pp. 6016, 2021.

[43] A. Alsufyani, Y. Alotaibi, A. O. Almagrabi, S. A. Alghamdi and N. Alsufyani, "Optimized intelligent data management framework for a cyber-physical system for computational applications," *Complex & Intelligent Systems*, vol. 10, pp. 1–13, 2021.

[44] H. H. Khan, M. N. Malik, R. Zafar, F. A. Goni, A. G. Chofreh *et al.,* "Challenges for sustainable smart city development: A conceptual framework," *Sustainable Development*, vol. 28, no. 5, pp. 1507–1518, 2020.

[45] G. Suryanarayana, K. Chandran, O. I. Khalaf, Y. Alotaibi, A. Alsufyani *et al.,* "Accurate magnetic resonance image super-resolution using deep networks and Gaussian filtering in the stationary wavelet domain," *IEEE Access*, vol. 9, pp. 71406–71417, 2021.

[46] G. Li, F. Liu, A. Sharma, O. I. Khalaf, Y. Alotaibi *et al.,* "Research on the natural language recognition method based on cluster analysis using neural network," *Mathematical Problems in Engineering*, vol. 2021, pp. 13, 2021.

[47] U. Srilakshmi, N. Veeraiah, Y. Alotaibi, S. A. Alghamdi, O. I. Khalaf *et al.,* "An improved hybrid secure multipath routing protocol for MANET," *IEEE Access*, vol. 9, pp. 163043–163053, 2021.

[48] N. Veeraiah, O. I. Khalaf, C. V. P. R. Prasad, Y. Alotaibi, A. Alsufyani *et al.,* "Trust aware secure energy efficient hybrid protocol for manet," *IEEE Access*, vol. 9, pp. 120996–121005, 2021.

[49] Y. Alotaibi, "A new secured E-government efficiency model for sustainable services provision," *Journal of Information Security and Cybercrimes Research*, vol. 3, no. 1, pp. 75–96, 2020.

[50] V. Ramasamy, Y. Alotaibi, O. I. Khalaf, P. Samui and J. Jayabalan, "Prediction of groundwater table for Chennai region using soft computing techniques," *Arabian Journal of Geosciences*, vol. 15, no. 9, pp. 1–19, 2022.

[51] S. Sennan, K. Gopalan, Y. Alotaibi, D. Pandey and S. Alghamdi, "EACR-LEACH: Energy-aware cluster-based routing protocol for WSN based IoT," *Computers, Materials & Continua*, vol. 72, no. 2, pp. 2159–2174, 2022.

[52] S. R. Akhila, Y. Alotaibi, O. I. Khalaf and S. Alghamdi, "Authentication and resource allocation strategies during handoff for 5G IoVs using deep learning," *Energies*, vol. 15, no. 6, pp. 2006, 2022.

[53] P. Kollapudi, S. Alghamdi, N. Veeraiah, Y. Alotaibi, S. Thotakura *et al.,* "A new method for scene classification from the remote sensing images," *CMC-Computers, Materials & Continua*, vol. 72, no. 1, pp. 1339–1355, 2022.

[54] U. Srilakshmi, S. Alghamdi, V. V. Ankalu, N. Veeraiah and Y. Alotaibi, "A secure optimization routing algorithm for mobile ad hoc networks," *IEEE Access*, vol. 10, pp. 14260–14269, 2022.

[55] S. Palanisamy, B. Thangaraju, O. I. Khalaf, Y. Alotaibi and S. Alghamdi, "Design and synthesis of multi-mode bandpass filter for wireless applications," *Electronics*, vol. 10, pp. 2853, 2021.

[56] S. Sennan, D. Pandey, Y. Alotaibi and S. Alghamdi, "A novel convolutional neural networks based spinach classification and recognition system," *Computers, Materials & Continua*, vol. 73, no. 1, pp. 343–361, 2022.

[57] A. R. Khaparde, F. Alassery, A. Kumar, Y. Alotaibi, O. I. Khalaf *et al.,* "Differential evolution algorithm with hierarchical fair competition model," *Intelligent Automation & Soft Computing*, vol. 33, no. 2, pp. 1045–1062, 2022.

[58] H. S. Gill, O. I. Khalaf, Y. Alotaibi, S. Alghamdi and F. Alassery, "Fruit image classification using deep learning," *Computers, Materials & Continua*, vol. 71, no. 3, pp. 5135–5150, 2022.

[59] H. S. Gill, O. I. Khalaf, Y. Alotaibi, S. Alghamdi and F. Alassery, "Multi-model CNN-RNN-LSTM based fruit recognition and classification," *Intelligent Automation & Soft Computing*, vol. 33, no. 1, pp. 637–650, 2022.

[60] H. H. Khan, M. N. Malik, Y. Alotaibi, A. Alsufyani and S. Alghamdi, "Crowdsourced requirements engineering challenges and solutions: A software industry perspective," *Computer Systems Science and Engineering*, vol. 39, no. 2, pp. 221–236, 2021.

[61] M. N. Malik, H. H. Khan, A. G. Chofreh, F. A. Goni, J. J. Klemeš *et al.,* "Investigating students' sustainability awareness and the curriculum of technology education in Pakistan," *Sustainability*, vol. 11, no. 9, pp. 2651, 2019.

[62] J. Tariq, A. Alfalou, A. Ijaz, H. Ali and I. Ashraf *et al.,* "Fast intra mode selection in HEVC using statistical model," *Computers, Materials and Continua*, vol. 70, no. 2, pp. 3903–3918, 2022.

[63] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion *et al.,* "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011; R. Meenakshi, R. Ponnusamy, S. Alghamdi, O. I. Khalaf and Y. Alotaibi, "Development of mobile app to support the mobility of visually impaired people," *Computers, Materials & Continua*, vol. 73, no.2, pp. 3473–3495, 2022.

[64] P. Tomar, G. Kumar, L. P. Verma, V. K. Sharma, D. Kanellopoulos *et al.,* "CMT-SCTP and MPTCP multipath transport protocols: A comprehensive review," *Electronics*, vol. 11, no. 15, pp. 2384, 2022.

[65] S. S. Rawat, S. Singh, Y. Alotaibi, S. Alghamdi and G. Kumar, "Infrared target-background separation based on weighted nuclear norm minimization and robust principal component analysis," *Mathematics*, vol. 10, no. 16, pp. 2829, 2022.

[66] Y. Alotaibi and F. Liu, "A novel secure business process modeling approach and its impact on business performance," *Information Sciences*, vol. 277, pp. 375–395, 2014.