IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 19, 2021, accepted March 15, 2021, date of publication March 26, 2021, date of current version April 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3069001

An Enhanced Hybrid Feature Selection Technique Using Term Frequency-Inverse Document Frequency and Support Vector Machine-Recursive Feature Elimination for Sentiment Classification

NUR SYAFIQAH MOHD NAFIS[1] **AND SURYANTI AWANG**[1,2]

[1]Soft Computing and Intelligent Systems Research Group (SPINT), Faculty of Computing, Universiti Malaysia Pahang, Pekan 26600, Malaysia
[2]Centre for Data Science and Artificial Intelligence (Data Science Centre), Universiti Malaysia Pahang, Kuantan 26300, Malaysia

Corresponding author: Suryanti Awang (suryanti@ump.edu.my)

This work was supported in part by the Ministry of Higher Education of Malaysia through the Fundamental Research Grant Scheme under Grant FRGS/1/2019/ICT02/UMP/02/1.

ABSTRACT Sentiment classification is increasingly used to automatically identify a positive or negative sentiment in a text review. In classification, feature selection had always been a critical and challenging problem. Most of the related feature selection for sentiment classification techniques unable to overcome problems of evaluating the significant features that will reduce the classification performance. This paper proposes an enhanced hybrid feature selection technique to improve the sentiment classification based on machine learning approaches. First, two customer review datasets namely Sentiment Labelled and large IMDB are retrieved and pre-processed. Next, the proposed feature selection technique which is the hybridization of Term Frequency-Inverse Document Frequency (TF-IDF) and Supports Vector Machine (SVM-RFE) is developed and tested on these two datasets. TF-IDF aims to measure features importance. The SVM-RFE iteratively evaluates and ranks the features. For sentiment classification, only the *k*top features from the ranked features will be used. Finally, the Support Vector Machine (SVM) classifier is deployed to observe the performance of the proposed technique. The performance is measured using accuracy, precision, recall, and F-measure. The experimental results show promising performances with 84.54% to 89.56% in the measurements especially from the large IMDB dataset. The results also outperformed other related techniques in certain datasets. Consequently, the proposed technique able to reduce from 19.25% to 70.5% of the features to be classified. This reduction rate is significant in optimally utilizing the computational resources while maintaining the efficiency of the classification performance.

INDEX TERMS Sentiment classification, sentiment analysis, text classification, product review, computational intelligence.

I. INTRODUCTION

can now provide feedback and reviews about a product digitally via social media or an online platform. This massive data is useful in business analytics, but it must be processed in a specific way. This method is referred as sentiment classification. Sentiment classification is defined as the entire process of extracting and comprehending emotion, opinion, and feedback expressed in text documents, whether positive, negative, or neutral. Customers' reviews, ratings, feedbacks, and comments are all examples of electronic word-of-mouth (e-WOM). Sentiment classification via e-WOM is currently popular. It is one of the most important factors influencing consumer behaviour [1]. It also has an impact on business decisions. As a result, the sentiment classification for product reviews should be highlighted. The product reviews are in the form of an unstructured text, which usually requires a detailed processing task to extract useful sentiments from it. It is due to the text document's non-uniform format, which includes unwanted words such as stop-words, symbols, and sometimes URLs.

In sentiment classification, there are two approaches: lexicon-based and machine learning-based (ML). The lexicon-based approach measures sentiment based on the semantic orientation of the text document, and it is dictionary-dependent. As a result, inadequately defined linguistic

The associate editor coordinating the review of this manuscript and approving it for publication was Wai-Keung Fung.

resources will contribute to poor classification performance. In comparison, the ML approach, which is non-dictionary dependent, uses any ML predictive performance to identify the sentiment. It makes this approach simple to implement and maintain. It was also proved that ML-based sentiment classification performed better [2]. However, both approaches require the use of a feature selection technique to identify the significant feature for classification.

There are three approaches to feature selection in ML-based systems: filter, wrapper, and hybrid. The filter approach evaluates feature importance using feature relevance metrics such as Chi-Square, Mutual Information, and Odds Ratio. The wrapper, on the other hand, is claimed to be a classifier-dependent approach that selects features based on the ML's predictive performance of a classifier on a given subset [3]. It takes time because of the learning algorithm process in feature selection. The hybrid feature selection approach combines any filter and wrapper feature selection approaches that are aimed at overcoming the drawbacks of the filter and wrapper techniques. The optimal hybrid feature selection approach combined the benefits and drawbacks of the filter and wrapper feature selection approaches. Nonetheless, the hybrid approach yielded the best classification results [4]. Furthermore, the size of features has a significant impact on ML's performance. Hence, the hybrid feature selection is the best approach to overcome the drawbacks of filter and wrapper approach.

Integrating two filter feature selection technique failed to give significant performance due to the absence of a ML algorithm. For example, integrating Chi-Square to re-evaluate the features evaluated by the TF-IDF [5]. There is a study that integrates TF-IDF with Next Word Negation to deal with word negation issues, but it leads to an increase in the number of features to be evaluated [6]. As a result, this paper proposes an enhance hybrid feature selection technique to utilize on the benefits of ML-based sentiment classification. It is designed to overcome the limitations of existing feature selection techniques. Motivated by the work of Luo & Luo, who used Support Vector Machine-Recursive Feature Elimination (SVM-RFE) to re-evaluate Odd Ratio rated features for Chinese text classification [7], we propose an enhanced hybrid of the TF-IDF and SVM-RFE for feature selection. The TF-IDF as a filter feature selection technique will evaluate the relevance of features in the text document, and the SVM-RFE will rank the features by selecting the significant features.

The rest of this paper is structured as follows: Section 2 presents related works about feature selection techniques. Meanwhile, Section 3 describes the proposed framework. The experimental setup and evaluations of the state-of-the-art techniques are presented and discussed in Section 4. Finally, Section 5 summarizes the conclusions and future works.

II. RELATED WORKS

We review the approaches used in feature selection for sentiment classification in previous works. To begin, we present a number of surveys and comparative studies on sentiment classification. Second, we reviewed previous research on feature selection for sentiment classification.

A. THE MACHINE LEARNING-BASED SENTIMENT CLASSIFICATION

Previous research's surveys and comparative studies will aid in identifying sentiment classification trends.

Ahmad *et al.* found that SVM is one of the most widely used ML techniques for detecting polarity from text documents in their systematic literature review [8]. In addition, they stated that, along with traditional ML classification techniques, researchers proposed customised and hybrid techniques. However, the scope of this study is limited to critical reviews of the literature on sentiment analysis using SVM from 2012 to 2017. Besides, Drus & Khalid conclude in their systematic literature review that SentiWordnet and TF-IDF are the most common Lexicon-based approaches for sentiment analysis, while Naive Bayes and SVM are for ML-based approaches [9]. The study examines studies published between 2014 and 2019.

Following that, Hameed *et al.* carried out an empirical study on sentiment classification techniques [10]. Six machine learning algorithms were tested on various sentiment benchmark datasets: Naive Bayes, Bagging, Random Forest, Decision Tree, Support Vector Machine, and Maximum Entropy (ME). The SVM classifier shows significant performance among other classifiers with the highest accuracy and lowest error rate However, the accuracy and error rate were only considered in this study to assess the classifier's capability, and no feature selection technique was mentioned.

According to Kumar & Rajini's survey, feature selection using mathematical or statistical-based approach is simple to use with a classifier because the implementation is straightforward and dictionary-independent [11]. The most common statistical techniques studied for feature selection are Chi-Square (Chi), Information Gain (IG), Odd-Ratio (OR), and Sequential Minimal Optimization (SMO). The k-Nearest Neighbor (NB) and Nave Bayes (NB) classifiers are also evaluated. However, only the NB classifier outperformed the other classifiers when the training set increment number was increased.

Tripathi & S presented a hybrid of natural language processing and machine learning techniques [12]. Term Occurrence, Binary term Occurrence, Term Frequency, and TF-IDF are part of the word vector creation process (Term Frequency-Inverse Document Frequency). The proposed technique was tested on the movie sentiment polarity dataset. Among these three-word vectors, term occurrence and binary term occurrence performed best for the Nave Bayes classifier, while TF-IDF performed best for the SVM classifier. They extended the experiment by employing N-grams. However, the classification performance did not significantly improve from 84.75% to 86.00%.

Meanwhile, Sahu & S. Ahuja conducted research on feature selection and classification for sentiment analysis [13].

A large IMDB movie review dataset with 50000 samples is tested to classify the polarity of the movie review on a scale of 0 (extremely disliked) to 4 (highly liked. The research focuses on feature extraction and selection, SentiWordNet, and Information Gain (IG).It identifies ten sentiment categories in feature extraction using SentiWordNet and N-gram techniques. The IG score and feature ranking algorithm were used to evaluate each feature's impact on the polarity of the document. To predict the class label, they used the Bagging, Random Forest, Decision Tree, Naive Bayes, K-Nearest Neighbor, and Classification via Regression classifiers. The study concludes that the proposed techniques outperformed state-of-the-art techniques with an accuracy of 88.95 %. However, the use of SentiWordNet indicates the dependency on a language-specific dictionary, which causes feature extraction and selection to take longer.

Avinash & Sivasankar studied the performance of TF-IDF (Term Frequency-Inverse Document Frequency) and Doc2vec (Document to Vector) as feature extraction techniques [14]. These two feature extraction techniques are trained and tested on three benchmark datasets with the classifiers Logistic Regression, SVM, k-NN, Decision Tree, and Bernoulli Naive Bayes. They concluded that Doc2vec and TF-IDF performed well on the majority of the datasets. Logistic regression, SVM with linear and RBF kernels, and RBF classifiers outperformed all other classifiers tested.

Following that, AlSaffar *et al.* [15] investigate how feature selection can improve Malay sentiment classification performance. Automatic sentiment classification experiments on online Malay-written reviews are carried out using three supervised ML classifiers and seven feature selection techniques. The experimental results showed that feature selection improves Malay sentiment classification. The combination of SVM as feature selection and classifier yielded the highest accuracy of 87%. In addition, when compared to NB and KNN, SVM produces more accurate results for medium and large feature sets.

Al Amrani *et al.* investigated and proposed Random Forest (RF) and SVM as hybrid sentiment analysis methods [16]. On the Amazon dataset, they tested and compared RF, SVM, and hybridization of RFSVM. They used the advantages of both RF and SVMs to solve the classification problem and improve classification performance. Using the RFSVM technique, they achieved an accuracy of 83.4 %.

Finally, Madasu and Elango [17] experiment with various feature selection techniques on Sentiment Labeled dataset. The selected features are trained using a variety of machine learning classifiers, including Logistic Regression (LR), Support Vector Machines (SVM), Decision Tree (DT), and Naive Bayes (NB). The base classifiers are those machine learning classifiers. To improve the efficiency of sentiment analysis, classifiers are trained using Ensemble Bagging and Random Subspace techniques. The experimental results show that, of both the feature selection techniques tested, Chi-Square and Count Difference produced the best results. Multinomial Naive Bayes and Logistic Regression perform excellently among base classifiers. When compared to Bagging, Random Subspace outperformed Bagging in Ensemble techniques.

From the previous studies on sentiments classification mentioned above, we concluded that ML is the most frequent approach used due to it's effectiveness. And SVM is the most frequent classifier tested. It gave competitive classification performances. Besides, various study had been made on feature selection for sentiment analysis. To conclude, the classification performance depends on the dataset's quality and size, and the feature selection used.

B. THE FEATURE SELECTION FOR SENTIMENT CLASSIFICATION

We looked at previous studies on hybrid feature selection for sentiment classification in this section. These will assist in gaining a better understanding of the issues that have arisen as a result of the existing techniques. When compared to the filter and wrapper approach, the hybrid feature selection approach is advantageous in terms of reducing the number of features desired to simplify the machine learning process [4], [18], [19]. Several studies on filter and wrapper feature selection reported lower performance measures [4] [20], [21].

A previous study on product review sentiment classification proposed by Bhuvaneswari & Parimala employed hybrid feature selection; namely, Sentiment Reviews Classification using Hybrid Feature Selection (SRCHFS). To improve classification performance, it extracts Synsets feature set with Correlation feature selection [22]. There were three types of product review datasets tested: movie review, multi-domain, and Amazon. The first movie review dataset achieves the highest accuracy of 94%, while the multi-domain dataset achieves 91% accuracy for the DVD domain. Finally, it was tested on two Amazon dataset domains: Cellphone and Restaurant, and it achieved 86.0 % and 83.5 %, respectively. However, in this study, accuracy is used solely as a performance metric. Multiple criteria, such as precision and recall, should be used to evaluate classification performance.

Dey *et al.* used TF-IDF and Next Word Negation (NWN) to classify sentiments [23]. The NWN is used to handle common word negations such as 'No', 'Not' and 'Never' found in any English text document. It is considered as stop-word and will usually remove during the pre-processing task. The proposed technique was tested on three datasets: Movie Review, Product Review, and SMS Spam, and it used a variety of classifiers, including Linear Support Vector Machine (LSVM), Multinomial Naive Bayes (MVB), and Max Entropy Random Forest (MERF). This study obtained 96% accuracy. The NWN's involvement, on the other hand, increased the number of features in the feature set.

Larasati *et al.* attempted to improve sentiment classification accuracy in a movie review dataset using SVM as a classifier—while also using Chi-Square Statistic and TF-IDF for feature selection [5]. Without the feature selection, the accuracy was only 68.7%. Then, with 80.2% accuracy, it shows a significant improvement with the proposed feature selection techniques. Nonetheless, it only assesses a

TABLE 1. Sentiment Dataset.

Class Label	Sentimetn Labelled Category			Large IMDB
	IMDB	Yelp	Amazon	
Positive	496	500	500	4929
Negative	504	500	500	5071
Total	1000	1000	1000	10000

classification model's accuracy as sentiment classification performance, which is insufficient. Multiple criteria, such as precision and recall, should be used to evaluate classification performance.

Iqbal *et al.* [24] proposed a feature reduction technique for sentiment classification based on the Genetic Algorithm (GA).The proposed technique was tested on three sentiment datasets: the UCI ML dataset for sentiment scoring, the Twitter labelled sentiment analysis dataset, and the geopolitical dataset related to the 2016 US Presidential Election. The non-GA and GA-based feature selection performed equally well on the UCI ML dataset. They claimed, however, that GA optimization could reduce the original feature set.

Based on these related works, it is possible to conclude that the majority of the studies have used various feature selection techniques to improve classification performance. The less related research, on the other hand, looked at the ML-based approach in hybrid feature selection techniques, specifically the hybridization of filter and wrapper approaches. Although there are studies that implemented the hybrid feature selection technique, it does not improve the performance significantly. It is because, most of the techniques did not focus on measuring the features importance using machine learning algorithm when selecting the significant features to be classified.

As we can see, this is crucial in order to achieve a good performance with fewer features to classify. As a result, it is critical to develop an effective feature selection technique that can meet the needs. As a result, we proposed combining the TF-IDF and SVM-RFE as an enhanced hybrid feature selection methods for sentiment classification. The TF-IDF will determine the importance of a feature in a text document, and the SVM-RFE will re-evaluate and rank the features by selecting the significant features using SVM score and recursive feature elimination.

III. OVERVIEW OF THE PROPOSED TECHNIQUE

The proposed hybrid feature selection technique consists of two stages. It aims to improve sentiment classification performance in order to address issues that have arisen in existing techniques, as explained in the preceding section. It consists of TF-IDF and SVM-RFE, the filter and wrapper feature selection approaches. The specifics of the procedure will be explained in the following section.

The framework of this study is depicted in Figure 1. It is divided into five stages: data collection, pre-processing, feature selection, classification, and evaluation of the proposed technique. The main focus is on the phase of feature selection.

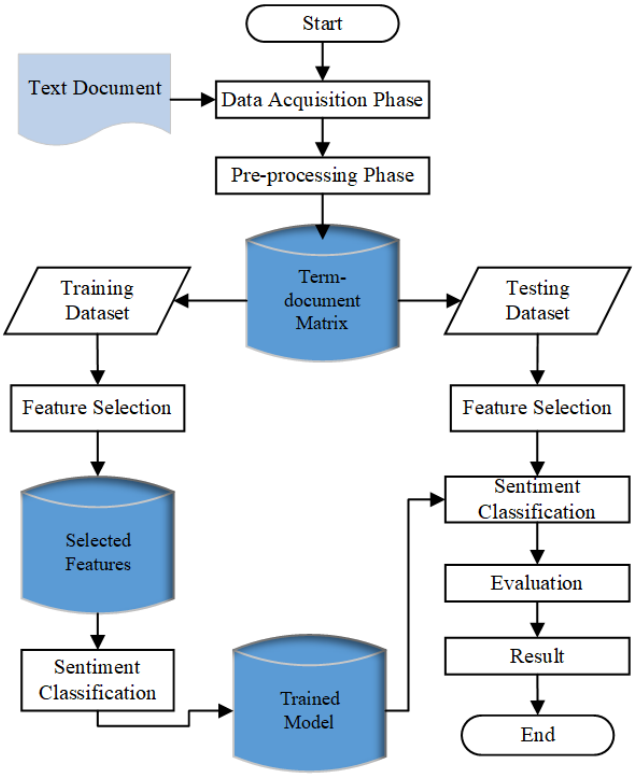

FIGURE 1. The General Methodology Framework.

This research framework begins with the sentiment review data collection in the data acquisition phase. The retrieved dataset will be pre-processed, which will include a few tasks. A term-document matrix (TDM) will be generated during the pre-processing phase. The TDM is divided into training and testing datasets as this is a supervised research framework. The proposed feature selection method is applied to both training and testing datasets. The selected features will then be trained to generate a trained classification model for the testing dataset. Finally, the classification result is obtained and assessed.

A. DATA ACQUISITION PHASE

Two benchmark datasets from the Kaggle website are used to evaluate the proposed technique on different dataset sizes. The datasets are a sentiment labelled dataset (known as Sentiment Labelled) and an Internet Movie Database movie review dataset (known as Large IMDB). Table 1 summarises the datasets' details. The Sentiment Labelled dataset is divided into three categories: Amazon, Yelp, and IMDB movie reviews. It includes 1000 samples in each category, with a balanced number of positive and negative sentiment reviews.

The second dataset is obtained from a large IMDB collection. The IMDB dataset is used to test the effectiveness of the proposed feature selection technique on a larger dataset. This dataset originally contained 50000 samples; however, due to computational resource constraints, we randomly selected 10000 samples for experimentation. The data distribution for

TABLE 2. The Example of Sentiment Text.

Text	Class Label
A very, very, very slow-moving, aimless movie about a distressed, drifting young man.	0
This short film certainly pulls no punches.	0
The movie showed a lot of Florida at its best, made it look very appealing.	1
The structure of this film is easily the most tightly constructed in the history of cinema.	1

TABLE 3. Large IMDB.

Class Label	Number of Samples
Positive	4929
Negative	5071
Total	1000

positive and negative is 4929 and 5071, respectively. The datasets are split into 2 parts: 80% for training and 20%t for testing. Table 2 shows a sentiment text document example. The text from 1 to 4 represents the review samples, for example. The class label '0' denotes a negative review, while '1' denotes a positive review.

B. PRE-PROCESSING PHASE

Text documents are always in an unstructured form whereby it is difficult to extract the hidden information. A readable format for the classifier is required, and pre-processing is needed. The aim of the pre-processing is more than to clean the text. It helps extract the text features since several symbols, URLs, and words may not be useful for the classification. In the pre-processing phase, it includes several tasks to convert unstructured text documents into a word vector as follows:

- Tokenization
- Stop-word removal
- Short-word removal
- Stemming
- Term-document matrix (TDM) conversion
- **Tokenization:** Tokenization chunked text sentences into meaningful words called tokens. Using whitespaces, the text document was chunked from a long paragraph. Furthermore, the HTML tag, XML scripts, special characters, and punctuation in a text document have no effect on performance. As a result, removing them helps to reduce the number of features in the classification phase. Figure 2 depicts the input text before the tokenization and the output after the tokenization task.
- **Stop-word Removal:** This task aims to remove tokens or terms from the text document that are commonly referred to as 'functional words' because they have no significant meaning, such as 'this', 'is', and 'but'. Because stop-words are repeated in a text, they have no effect on the classification process and reduce computational complexity. This study employs the SMART stop-list, which contains a list of commonly used stop-words in the English language [25].

Algorithm 1 SWORD_REMOVAL

Input:Array of words, $T = t_1, t_2, t_3 \ldots$until t_n
Output:Array of new words, $U = u_1, u_2, u_3 \ldots$until u_n
S_W: Stop-word list.
1: Read T and S_W.
2: **for** t_n **do** 3 to 6
3: Compare the S_W to T using the sequential search.
4: **if** $t_n = S_W$
5: Remove t_n from T
6: **end if**
7: Update array: *Update* $U = T$
8: End

The stop-word removal algorithm was summarised in Algorithm-1. It starts by putting the tokens into an array. Then it reads each stop-word in the SMART stop-word list one by one. The stop-word is then compared to the token. If it matches, the token is removed. This process is repeated until the end of the array. Furthermore, it will continue reading and comparing the stop-word until it reaches the end of the stop-word list. Finally, the new token array is obtained. Figure 3 depicts an example of input and output text following the stop-word removal task. Stop-words like 'this' and 'a' are removed from the input text. It generates new output text with fewer words for the next text pre-processing task.

- **Short-word Removal:** A short word with a length of 1 to 2 characters is omitted to reduce the number of tokens or features. These short words are frequently derived from short-form words, such as 'tq' (thank you) and 'ok' (read: okay). It is considered noise because grammatical and spelling errors are repeated in an informal text. As a result, it is recommended to remove from the text document in order to reduce the number of features.
- **Stemming:** The goal of stemming is to find the root word by removing suffixes. In the feature space, different tokens that share the same root-word can be identified as the same token. The number of tokens will be reduced as a result. The terms 'continued' and 'continuing', for example, share the same root word, 'continue'. Furthermore, singular, and plural words are stemmed. For example, the term 'levels' is stemmed to 'level'. Porter's algorithm for English sentences is used in this paper [26]. Because of its effectiveness and relative accuracy, Porter stemmer is widely used for many text classifications purposes [27]. Porter's algorithm

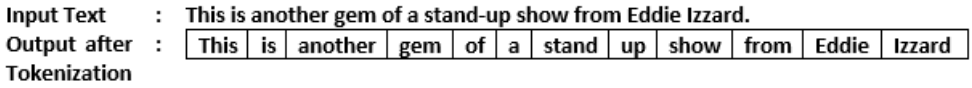

FIGURE 2. The example of tokenizing sentence.

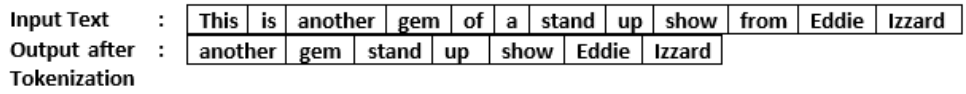

FIGURE 3. The example of input and output for the stop-word removal task.

TABLE 4. The Example of TDM.

Document, D_N	Feature, F_N								
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9
D_1	0	0	1	0	1	0	1	0	0
D_2	1	0	0	0	0	0	0	1	1
D_3	1	0	0	2	0	1	0	1	0
D_4	0	1	0	0	1	0	1	0	1

employs a five-step procedure that can be applied to any input text. A term is specified as $[C](VC)m[V]$ according to the algorithm, where C and V are lists of consonants and vowels respectively, and m is the term's scale. It employs sixty laws divided into five steps to accurately determine the stem of a term.

- **Term-document Matrix (TDM) Conversion:** The final pre-processing task is TDM conversion, with the goal of creating a classifier-readable format dataset. Each distinct token from the previous pre-processing is regarded as a feature. The frequency of each feature that appears in the document is recorded in TDM. TDM is illustrated in Table 3. TDM will be used in the next phase of feature selection and sentiment classification. The feature is denoted by F_N ($F_1 \ldots F_N$), which represents the number of features extracted during the pre-processing phase. Meanwhile, the document is represented by D_N (D_1 to D_4), which is the text document number. The frequency of each feature is represented by the value in the table. Note that frequency refers to the number of times the feature appears in the text document. For example, the frequency with which F_3 appears in D_1 is one; the frequency with which F_4 appears in D_3 is two.

C. FEATURE SELECTION

The feature selection technique is designed to reduce the number of irrelevant features and select significant features prior to the classification phase to improves the sentiment classification efficiency. The selected features will be used in the classification phase. Figure 4 illustrates the proposed feature selection technique framework. The stage 1 is non-predictive measure, and the stage 2 is predictive measure. Stage 1 is referred to as a non-predictive measure because it does not use a learning algorithm to select the feature. It only involved the TF-IDF calculation and related processes. whereas the predictive measure implements SVM-RFE. SVM-RFE, on the other hand, used the SVM learning algorithm as part of the feature selection process in the second stage.

As shown in table 4, the process begins in the non-predictive measure stage by calculating the TF-IDF for the term-document matrix. This calculation is used to determine the importance of a feature in a text document. The TF-IDF matrix's variance score is then computed, as shown in table 5. The variance score indicates the spread of the data from the mean. It is to determine which features are far from the mean. Furthermore, it is set as a threshold for selecting the TF-IDF score for the features. Referring to the table 5.The variance score obtained for the entire TF-IDF matrix is 0.180. In D_1, the TF-IDF score for F_3 is 0.175. As a result, the TF-IDF score for this feature is discarded because it is less than the variance score. At the end of this stage, it generates a new TF-IDF matrix with a lesser TF-IDF score for each feature.

In the stage 2, the SVM-RFE will be implemented on the new feature set to produce SVM-RFE ranked list. It is applied to re-evaluate feature importance using the SVM classifier and RFE algorithm. The SVM works as a classifier and RFE algorithm is a feature selection method that fits the training model by discarding the weakest feature(s) until the desired number of features that give high performance is reached. The process of selecting features started by defining several numbers of k-top features. The performance of the k-top features will be evaluated using the classifier. This process is repeating until there is no changes in the performances. Later, the k-top features with higher performance will be selected as the final result. The next sections will explain in detail each of the process.

1) TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY (TF-IDF)

The Term frequency-inverse document frequency (TF-IDF) score of the document is computed as the first step in this feature selection technique. The TF-IDF score indicates which feature is the most important in the whole document collection. The TF-IDF formula is deployed to the dataset. The

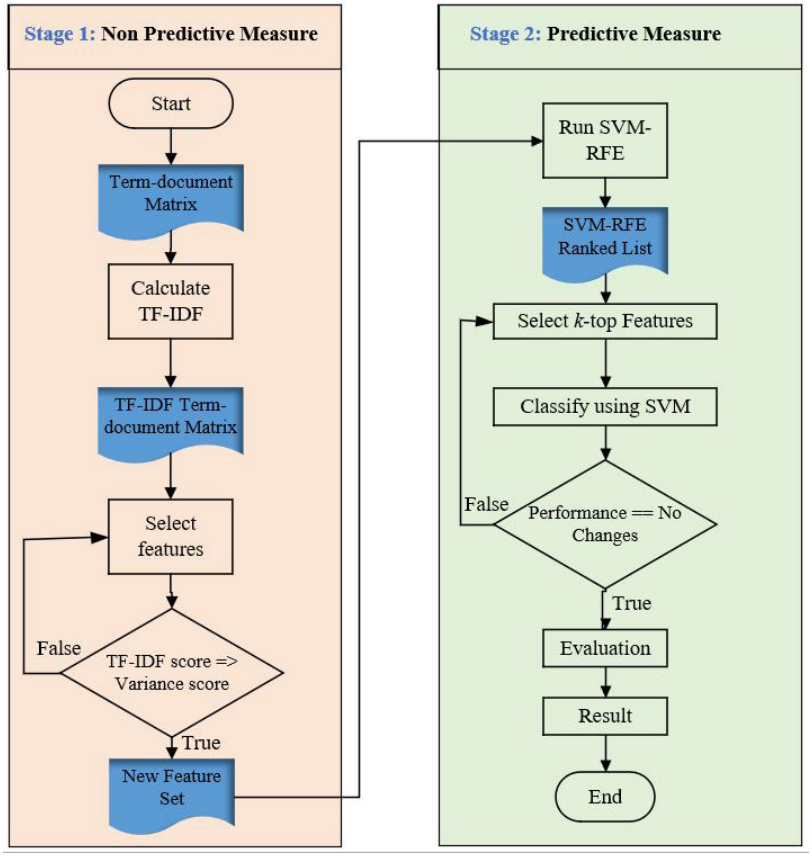

FIGURE 4. The proposed feature selection framework.

TF-IDF formula is written as follows:

$$TF - IDF = TF * IDF \tag{1}$$

TF defines as the frequency of a feature appears in a document over the total features appear in the text document. At the same time, IDF evaluates the ability of a feature to distinguish between categories. Note that, the categories here are the defined class label of the text document. TF and IDF are expressed in the following formula:

$$TF = \frac{FFTD}{TFTD} \tag{2}$$

$$IDF = \log\frac{NDF}{TD} \tag{3}$$

where, FFTD is the frequency of a feature appear in a text document, TFTD is the total number of a term appears in the text document. For the IDF, NDF is the number of document with the feature in them, and TD is the total number of document.

The higher a feature's TF-IDF score, the more important the feature is for a particular text document. Table 4 tabulates the previous term-document matrix's example TF-IDF matrix (TDM).

2) SUPPORT VECTOR MACHINE- RECURSIVE FEATURE ELIMINATION

SVM classifies binary class problems by identifying a separation between hyper-planes defined by classes of data. Assume that there is a given set, S, of points $x_i \in R^n$ with $i = 1, 2, 3 \ldots .n$. Each point x_i is belongs to either of two classes with a given label $y_i \in 0|1$.

The goal of this step is to create a hyper-plane equation that divides S while leaving all points of the same class on the same side. SVM commits classification by creating a N-dimensional hyperplane that divides the data into two categories optimally. The categories are the class label, namely positive or negative sample. SVM score, W can be written as the formula below:

$$W = \sum_{i=1}^{n} a_i y_i x_i \tag{4}$$

where, i is the number of terms ranging from 1 to n, a_i is Lagrangian multiplier estimated from the training set; x_i is term vector for sample i, and y_i is the class label. Weighted vector or SVM score defines the sum square of the weight vector W of the SVMs using the formula. The SVM score, W is used to evaluate the features.

Meanwhile, the recursive Feature Elimination is a feature selection method that can be applied to any training model,

TABLE 5. The Example of TF-IDF Score for TDM.

Document, D_N	Feature, F_N								
	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9
D_1	0	0	0.175	0	0.239	0	0.239	0	0
D_2	0.239	0	0	0	0	0	0	0.239	0.239
D_3	0.239	0	0	1.398	0	0.175	0	0.239	0
D_4	0	0.175	0	0	0.239	0	0.239	0	0.239

such as SVM. It discards the weakest feature (or features) until the desired number of removed features is reached. The training model ranks features by recursively removing a number of features per iteration. The goal is to eliminate any dependencies and collinearity that may exist in the model.

The SVM-RFE algorithm removes the irrelevant, redundant features, and noises in a sequential iterative process. The process of evaluating and eliminating features is determined by the parameters set at the start of the process, which are the remove ratio and the stop-chunk threshold. For example, the remove ratio parameter is set to 50%, and the stop chunk threshold is set to 100. It means that in each iteration, the algorithm will evaluate the features and remove 50% of them from the feature set. When the number of available features is less than 100, the algorithm will remove them one by one.

Algorithm-2 summarises the SVM-RFE algorithm. The initial feature subset is used as input. The initial subset, S, is chosen at random from the vector space. This algorithm returns R, which is a feature-ranked list based on the smallest weight criterion. This procedure is repeated until S is empty. The first step in obtaining the output is to assign an empty set, R. The SVM weight vector score is used to train the features in a linear SVM. It then computes the ranking criteria and sorts the results in descending order. The new feature rank has been updated. The elimination process begins by removing the features in the lowest rank that have the smallest SVM weight vector score. At the end of the process, a new feature list is created according to the smallest weight criterion. The smallest weight is the least important feature. It will be near the bottom of the list of features.

Algorithm 2 SVM-RFE

Input: Initial feature subset, $S = 1, 2, 3 \ldots . n$
Output: Ranked list according to the smallest weight criterion, R
W: Weight Vector
Rank: Ranking Criteria
1: Set $R = \{\}$
2: **If** $S =$ not $\{\}$, **do** 2 to 10
3: Train SVM using S.
4: Compute W using equation (4).
5: Compute $Rank = W^2$.
6: Sort rank; $Newrank = sort(Rank)$
7: Update rank; $UpdateR = R + S(Newrank)$
8: Eliminate the feature with the smallest rank;
$UpdateS = S - S(Newrank)$
10: **end if**
11: End

D. SENTIMENT CLASSIFICATION AND EVALUATION

In this section, we used an SVM classifier to assess the performance of the proposed technique. It should be noted that several classifiers are used in sentiment classification, including SVM, k-NN, Nave Bayes, and many others. However, the SVM is used in this study because the majority of related works have reported promising performance using this classifier compared to the other classifiers. Several performance measures, such as accuracy, precision, recall, and F-Measure scores, are used to evaluate classification performance.

1) SVM CLASSIFIER

Support Vector Machine (SVM) is a well-known machine learning algorithm that capable to handle high-dimensional data and achieved promising performance such as text document [28]–[30]. SVM is a non-probabilistic algorithm which able to separate data linearly and non-linearly. However, in solving the binary class problem, a linear SVM is sufficient. SVM classifier that we used in this classification is based on (4). SVM commits classification by constructing several N-dimensional hyper-planes that optimally split the data into two categories. Among the possible hyper-planes, the one where the distance of the hyper-plane from the closest data points (the ''margin'') is as large as possible is selected. It is indicating that it can assign data points to its correct class.

2) PERFORMANCE MEASUREMENT

Accuracy, precision, recall, and F-measure are the performance measures used to evaluate the effectiveness of each feature selection. The primary evaluation metric in classification is accuracy, which indicates how well a classification model predicts the class label for unknown samples. It is related to the human agreement baseline, which is around 80% to 85%. According to research, when evaluating the sentiments of a given text document, human analysts tend to agree around 80-85% of the time. It's known as the human baseline agreement [31]. Furthermore, in [32] the agreement on emotion categories in a text document ranges from 60% to 79%. Note that in (5) until (8), the i to N is samples, tp, tn, fn, and fp is true positive, true negative, false negative, and false positive respectively as in a confusion matrix. The accuracy is measured based on the number of correctly predicted over

the total number of samples. It can be formulated as in (5).

$$Accuracy = \sum_{i=1}^{N} \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i} \tag{5}$$

Next, precision is calculated by dividing the number of correctly classified samples by the total number of classified samples. Equation (6) calculates precision. Precision is measured to observe the number of correct positive prediction out of positive prediction. A higher precision score indicates that there are fewer false positives, whereas a lower precision score indicates that there are a lot of false positives. As a result, a higher precision score indicates better performance.

$$Precision = \frac{\sum_{i=1}^{N} tp_i}{\sum_{i=1}^{N} (tp_i + fp_i)} \tag{6}$$

The recall is defined as the proportion of correctly classified samples in relation to the total number of samples in a given class. It can be written as in (7). The recall is measured to observe the number of correct positive prediction out of the positive example. The higher the recall score, the fewer false negative predictions are predicted, and vice versa.

$$Recall = \frac{\sum_{i=1}^{N} tp_i}{\sum_{i=1}^{N} (tp_i + fn_i)} \tag{7}$$

Finally, the F-measure is introduced to avoid biassed evaluations of precision and recall. As shown in, it is a combination of precision and recall (8). F-measure is measured to seek a balance between Precision and Recall.

$$F - measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{8}$$

IV. EXPERIMENTS AND RESULTS

A. DATASET AND SETTINGS

The experiments are conducted using the Sentiment Labelled dataset and IMDB dataset, as mentioned in section 3.1 to classify the positive and negative sentiment. The Sentiment Labelled dataset is tested to evaluate the proposed techniques on the small benchmark dataset. Meanwhile, the IMDB dataset aims to test the proposed feature selection technique using a larger dataset with enormous features.

We divided the samples from these datasets into two groups: training and testing. The Sentiment Labelled database contains a total of 2400 samples, 800 for each category and 600 for the test dataset with 200 samples. The IMDB dataset has 8000 samples in the training dataset and 2000 samples in the testing dataset.

Experiments has been tested on an Intel i-7 platform on the MATLAB software. While utilizing SVM-RFE in the feature selection phase and sentiment classification, the linear SVM algorithm function is used. Besides, LIBSVM is adopted to solve the SVM model in the SVM-RFE algorithm. The 5-fold cross-validation is used with the parameters C is set to 1 in utilizing SVM. The algorithm is set to eliminate 50% of the feature at a time prior to the elimination of one feature at a time. We choose 50% as the feature elimination percentage to speed up the evaluation process.

TABLE 6. Performance Measurement for First Experiment using Sentiment Labelled Dataset.

Category	Performance Measurement (%)			
	Accuracy	Precision	Recall	F-measure
IMDB	82.66	80.98	83.75	82.34
Amazon	78.65	77.24	83.11	80.04
Yelp	84.14	87.41	83.78	85.55
Average	81.81	81.88	83.55	82.64

The results are observed based on two significant experiments. The first experiment is carried out without the proposed feature selection method, while the second experiment is carried out with it. The results obtained using the test data set are discussed in the following subsection. We also compare our proposed methodology to state-of-the-art techniques to see how well it performs.

B. SENTIMENT CLASSIFICATION WITHOUT THE FEATURE SELECTION

In the first experiment, we evaluated the classification without the feature selection technique. This experiment will serve as the baseline experiment for performance comparison in the following section. The processes for this sentiment classification begin with the acquisition of text documents, which is followed by the pre-processing step. Next, divide the dataset into a training dataset and a testing dataset, as shown in figure 1. We skipped over the feature selection process and went straight to the sentiment classification phase. All of the extracted features from the pre-processing are used in the classification step. The classification performance for the Sentiment Labelled dataset is summarised in Table 6.

The Yelp category outperformed other dataset categories in all performance measures for the Sentiment Labelled dataset, with 84.14%, 87.41%, 83.78%, and 85.55%, respectively. The Amazon category, on the other hand, performed the worst of the three. With an overall score of 81.81%, the classifier correctly classified the dataset's sentiment as positive and negative sentiment. The average precision score for this dataset is 81.88%, indicating that 81.88% of the estimated positive sentiment was correctly identified, and the average recall is 83.55%. Out of the actual positives, the average recall shows that 83.55% is correctly identified. Finally, the classifier received an average F-measure score of 82.64%.

Table 7 presents the classification results when this approach is extended to a larger dataset from IMDB. Due to resource constraints, we only picked 10000 samples from this dataset. We spent about 5 days on the entire sentiment classification process using these samples. In this experiment, the precision score is 82.24%, and the accuracy score is 83.16%. It correctly classified 83.70% of the actual positives in terms of recall. At the same time, the F-measure score obtained is 82.96%.

Overall, the results show that the classifier can identify sentiments using the respective performance measures without

TABLE 7. Performance Measurement for First Experiment using Large IMDB.

Performance Measurement (%)			
Accuracy	Precision	Recall	F-measure
83.16	82.24	83.70	82.96

TABLE 8. Number of The Extracted Features from Pre-processing.

Dataset	Number of Features
Sentiment Labelled (IMDB)	2039
Sentiment Labelled (Amazon)	1174
Sentiment Labelled (Yelp)	1355
Large IMDB	30959

TABLE 9. Performance Measurement for Second Experiment using Sentiment Labelled Dataset (IMDB Category).

k-top Features	Performance Measurement (%)			
	Accuracy	Precision	Recall	F-measure
200	76.33	82.03	66.04	73.13
400	78.49	82.07	70.94	76.08
600	78.84	82.07	70.94	76.08
800	78.89	80.21	74.69	77.34
1000	79.85	81.60	75.21	78.26
1200	78.64	80.53	73.54	76.87
1400	78.24	79.04	74.79	76.82
1600	79.60	78.34	79.79	79.05
1800	80.55	78.80	81.67	80.21
2000	82.31	80.11	84.27	82.13

the use of feature selection. In the classification phase, however, we used all of the extracted features, as shown in Table 8. We can see from that table that too many features are used in relation to the dataset's size. The minimum features used in the classification are 1174 for the Sentiment Labelled (Yelp category) dataset and 30959 features for the large IMDB dataset. Since they all reached the human agreement baseline of 80% to 85%, all datasets show good classification performance with the original feature set.

Nonetheless, the computing resources have been wasted since the classification process often tests irrelevant features. To maintain classification efficiency with fewer features, a feature selection technique is needed. Furthermore, the curse of dimensionality is avoided by using a feature selection technique. Note that the curse of dimensionality is a common classification problem that can degrade the efficiency of the classification.

C. SENTIMENT CLASSIFICATION USING THE PROPOSED FEATURE SELECTION TECHNIQUE

The proposed feature selection technique was tested in the second experiment to observe its performance. TF-IDF scores are calculated for each feature in each text document in the first stage of the feature selection technique. The variance score of the term-document matrix is determined to assess the spreads of the TF-IDF score of the features. Features with a TF-IDF score lower than the variance score are removed. As a result, the number of features for the respected sample is decreased, and a new feature set is generated for each sample. The SVM-RFE was then used to re-evaluate the new TF-IDF matrix. Only *k*-top features will be used to test sentiment classification. Similarly to the previous experiment, we used SVM as the classifier to assess the performance of the proposed technique.

Tables 8 to 10 show the performance measurements for the proposed technique when tested with Sentiment Labelled Datasets for the categories IMDB, Amazon, and Yelp. The goal of separating the results for this dataset is to evaluate the proposed technique's performance based on the *k*-top features of each category. The *k*-top features are chosen after the feature selection stage 2 is completed, as stated in the methodology section. The *k*-top feature groups are selected based on the size of the features. The larger the feature size, the bigger the *k*-top range. We chose *k*-top features in increments of 200, 400, 600, and so on because performance measurements show a significant difference with that number of features. Furthermore, each group of *k*-top features will result in either no change, an increase, or a decrease in performance. It will determine which *k*-top features group is the most effective at classifying sentiments.

According to table 8, the best performance is obtained with 2000-top features for all performance measurements, which are 82.31% for accuracy, 84.77% for recall, and 82.13% for F-measure. However, the performance for the precision measure is 80.11%, which is 1.96% lower than the highest precision measure from the 400-top and 600-top features. To To conclude, the 2000-top feature is considered the best feature in this dataset for the IMDB category because it received the highest score for three performance measures, namely accuracy, precision, and F-measure. The obtained precision score is also comparable to the highest precision score. The proposed technique is considered precise because the precision score in each group of *k*-top features is consistently high, even with the lowest of 200-top features.

Figure 5 depicts the proposed technique's performance in the IMDB category. The results show that performance improves in direct proportion to the number of *k*-top features added. However, we are unable to test with more than 2000 features because the maximum number of features in the IMDB category is only 2039. Furthermore, increasing the number of *k*-top features will result in exceeding the original feature set.

Table 9 shows the performance of the proposed technique in the Sentiment Labelled dataset for the Amazon category. We chose *k*-top features in increments of 100, 200, 300, and so on because performance measurements show a significant difference with that number of features. Furthermore, each group of *k*-top features will result in either no change, an increase, or a decrease in performance. It will determine which *k*-top features group is the most effective at classifying sentiments. The best performance is obtained when the selected features are 700-top, with all percentage performance measures at 80.05%, 84.06%, 75.63%, and 79.61%, respectively, for accuracy, precision, recall, and F-measure.

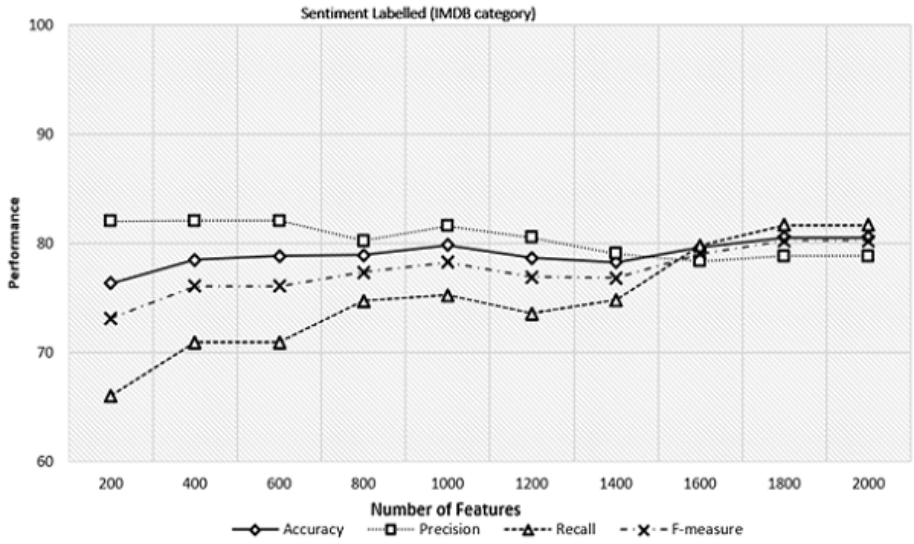

FIGURE 5. The performance of Sentiment Labelled Dataset (IMDB Category).

TABLE 10. Performance Measurement for Second Experiment using Sentiment Labelled Dataset (Amazon Category).

k-top Features	Performance Measurement (%)			
	Accuracy	Precision	Recall	F-measure
100	74.50	86.11	60.19	70.86
200	74.60	81.45	65.63	72.68
300	76.50	82.65	68.83	75.10
400	76.80	83.55	68.45	75.23
500	77.25	83.64	69.42	75.85
600	78.25	82.35	72.33	77.40
700	80.05	84.06	75.63	79.61
800	79.30	83.26	74.85	78.83
900	77.30	82.07	71.55	76.45
1000	79.00	82.31	75.44	78.72

TABLE 11. Performance Measurement for Second Experiment using Sentiment Labelled Dataset (Yelp Category).

k-top Features	Performance Measurement (%)			
	Accuracy	Precision	Recall	F-measure
100	83.79	87.76	82.61	85.10
200	83.84	87.98	82.43	85.12
300	83.64	87.58	82.52	84.97
400	84.04	88.03	82.79	85.33
500	83.54	87.99	81.80	84.78
600	83.64	87.87	82.16	84.91
700	83.74	88.18	81.98	84.97
800	83.59	87.78	82.16	84.88
900	83.28	87.42	81.98	84.61
1000	83.69	87.88	82.25	84.97
1100	83.69	87.66	82.52	85.01
1200	83.64	87.86	82.16	84.92

Furthermore, the proposed technique is considered precise because the precision score is consistently high in all *k*-top groups, even though it achieved 86.11% with the lowest 100-top feature. In contrast, the recall score in all *k*-top features is quite low, with no more than 80%. As a result, when the precision is high and the recall is low, it indicates that the proposed technique is sensitive to the true positive. All of the predicted true positives are true positives, but it also missed a lot of true positives. Figure 6 depicts the performance pattern of all Amazon performance measurements. It shows that performance increases in proportion to the increase in *k*-top features up to 700-top features. However, when the number of features is increased further than 700, the performance decreases in all performance measures.

Table 10 reveals the results of the proposed technique for the Yelp category. When 400 top features are selected for this category, the best classification performance is obtained. According to the results, the accuracy is 84.04 %, the recall is 82.79 %, and the F-measure is 85.33 %, the highest of any *k*-top feature group. The precision score in 400-top features, however, is 0.15% lower than the highest precision score, with 88.03% versus 88.18%. As a result, we considered that the precision score is comparable. To conclude, the 400-top feature has the best performance because it outperformed other *k*-top features group in all performance measures, despite a slightly lower precision percentage. Furthermore, from the lowest to the highest *k*-top features group show competitive results, demonstrating that the proposed technique achieved excellent results.

Figure 7 depicts the overall performance pattern of the Yelp category's performance measures. The results show that performance increases proportionally to the number of *k*-top features selected until 400-top features are selected. However, when the number of *k*-top features exceeds 400, the performance shows a static pattern, with the exception of the precision score. This scenario occurred as a result of the best feature set being met. When selecting 700-top features, however, it only shows a small increment.

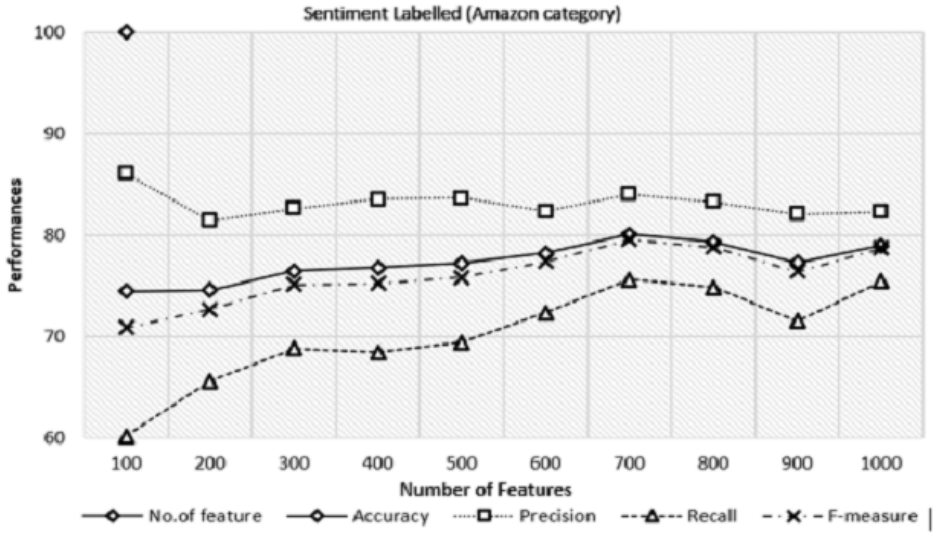

FIGURE 6. The performance of Sentiment Labelled Dataset (Amazon Category).

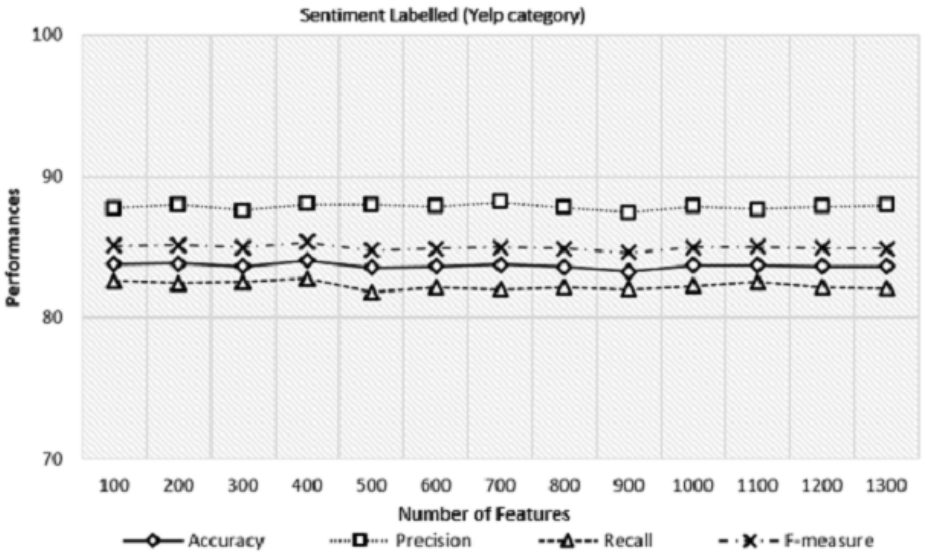

FIGURE 7. The performance of Sentiment Labelled Dataset (Yelp Category).

Following that, the proposed feature selection technique is tested on the large IMDB dataset. Note that the entire sentiment classification process took about five days to classify this dataset using the proposed technique. Table 11 shows the results of the performance measurements. Because we want to access the performance measurements, we chose *k*-top features of 500, 1000, 5000, 10000, and so on. It shows a significant difference with that number of features. In addition, each group of *k*-top features will produce either no change or increase or decrease in performance. It will determine which *k*-top features group is the best in classify the sentiments.

According to the table, selecting 25000-top features produced the highest accuracy score. The best precision score is 84.64 % from the 15000-top feature, and when the *k*-top is 30000, the recall and F-measure scores are 90.55 % and 87.04 %, respectively. However, the 25000-top features are considered the best classification performances because they demonstrate consistent comparable performance across all measurements. It also does not show a statistically significant difference when compared to the highest scores from other *k*-top features, particularly in precision, recall, and F-measure.

The precision score obtained is only 0.1% lower than the highest precision from the 15000-top features. For the recall score, it achieved 89.56%, which is only 0.99% lower compared to the 25000-top features with 90.55%. It obtained 86.96% for the F-measure score, which is only 0.08% lower

TABLE 12. Performance Measurement for Second Experiment using Large IMDB Dataset.

k-top Features	Performance Measurement (%)			
	Accuracy	Precision	Recall	F-measure
500	80.83	82.02	78.07	79.95
1000	83.77	82.15	85.50	83.75
5000	85.79	83.15	89.06	85.99
10000	86.44	84.62	88.47	86.47
15000	86.59	84.64	88.69	86.63
20000	86.72	84.11	89.93	86.90
25000	86.85	84.54	89.56	86.96
30000	86.80	83.86	90.55	87.04

compared to the 30000-top features with 87.04%. To conclude, the proposed technique is considered precise because the precision score is consistently high in each group of *k*-top features, even when the precision score is the lowest of 500-top features. Figure 8 shows that the accuracy and F-measure scores improve in direct proportion to the increase in *k*-top features. However, in the recall and precision scores, it shows that the performances are fluctuated across all the *k*-top features. Furthermore, when the *k*-top features is 30000, the classification performance degrades, particularly in terms of accuracy and precision. Note that the maximum number of features in this dataset is 30959.

Table 12 compares the highest performance measurements obtained from the first and second experiments. The first experiment is the baseline experiment in this paper, without the feature selection technique, and the second experiment is the proposed technique. Based on that table, we can see that the performance measurements for the Sentiment Labelled dataset in the IMDB category do not differ significantly. In all measurements, the results of both experiments were comparable. However, the proposed technique is unable to select fewer *k*-top features to achieve those scores, with 2000-top features selected from a total of 2039 features, representing only 2% of the feature reduction.

In the Amazon category, the proposed technique achieved 80.05% accuracy compared to the baseline experiment's 78.65%. Furthermore, 84.06% precision is achieved, compared to only 77.24% without feature selection. These results outperform the baseline experiment results when only 700 top features are classified as compared to 1174 features. However, the proposed technique has a slightly lower recall and F-measure score, with a difference of 7.48% and 0.43%, respectively. Nonetheless, we were able to reduce 40.4% of the features in this category in order to achieve higher performance measurements, particularly in accuracy and precision.

In the Yelp category, the proposed technique achieves 84.04 % accuracy compared to 84.14 % in the baseline experiment, a difference of only 0.1 %. In terms of precision, the proposed technique outperformed the baseline experiment by 88.03 % versus 87.41 %, a difference of 0.62 %. Aside from that, the recall and F-measure perform similarly in both experiments, with the baseline experiment having a higher percentage. The difference in percentage is only 0.9 % and 0.2 %, respectively. However, by employing the proposed technique, we were able to reduce 70.5 % of the features, requiring only 400 top features to achieve the same results as 1355 features.

Finally, we compared the performance measures based on the Large IMDB dataset. Looking at that table, we can see that when we implemented the proposed technique, we achieved promising results in all measurements. The accuracy is 86.85 % and the precision is 84.54 %, compared to the baseline experiment's 83.16 % and 82.24 %, respectively. In addition, the recall and F-measure were able to achieve very high performance with 89.56 % and 86.96 %, respectively, compared to only 83.70 % and 82.96 %. The most significant accomplishment is that we were able to reduce 19.25 % of the features in order to achieve these high performance levels. It is worth noting that the features deployed in this classification are 25000-top features as opposed to 30959 features.

In conclusion, the proposed feature selection technique reduces the number of features during the classification from 19.5% to 70.5% to achieve comparable and even better performances in certain dataset. As a result, it also assists in the reduction of computational complexity for classification. Furthermore, it maintains the human agreement baseline of 80% to 85% in the majority of the performances achieved, and at some points, the performances are even better than this baseline.

D. COMPARISON WITH OTHER STATE-OF-ART TECHNIQUES

We conducted a comparison of our proposed feature selection technique with other state-of-the-art techniques. In this comparison, non-hybrid and hybrid ML-based feature selection techniques are considered. The goal of this comparison is to evaluate the effectiveness of the proposed feature selection technique on sentiment classification performance. We only provided the Sentiment Labelled results for comparison because the compared techniques were also tested on this dataset. However, because the state-of-the-art techniques did not provide other performance measurements in their paper, only accuracy is considered in this comparison. They did not also use the large IMDB that we used in this paper. We are also unable to compare our results to those of other techniques that used a large IMDB dataset because the size of the dataset varies. There were papers that used 5000 samples due to resource constraints, and some papers used all 50000 samples because they did not have any resource constraints.

According to table 13, the proposed feature selection technique outperformed the state-of-the-art techniques in the IMDB and Yelp categories. The highest accuracy is 82.31% when the IMDB category is used, compared to 79.67% for work done by Madasu and Elango and 80.2% for Larasati *et al.* It is 2.64% and 2.11% higher than state-of-the-art techniques, respectively. The proposed technique obtained 2000 out of 2039 features for classification. However, the state-of-art studies do not specify the number of features obtained. We also had the highest accuracy in the Yelp category, with 84.04%, compared to 81.33% for Madasu

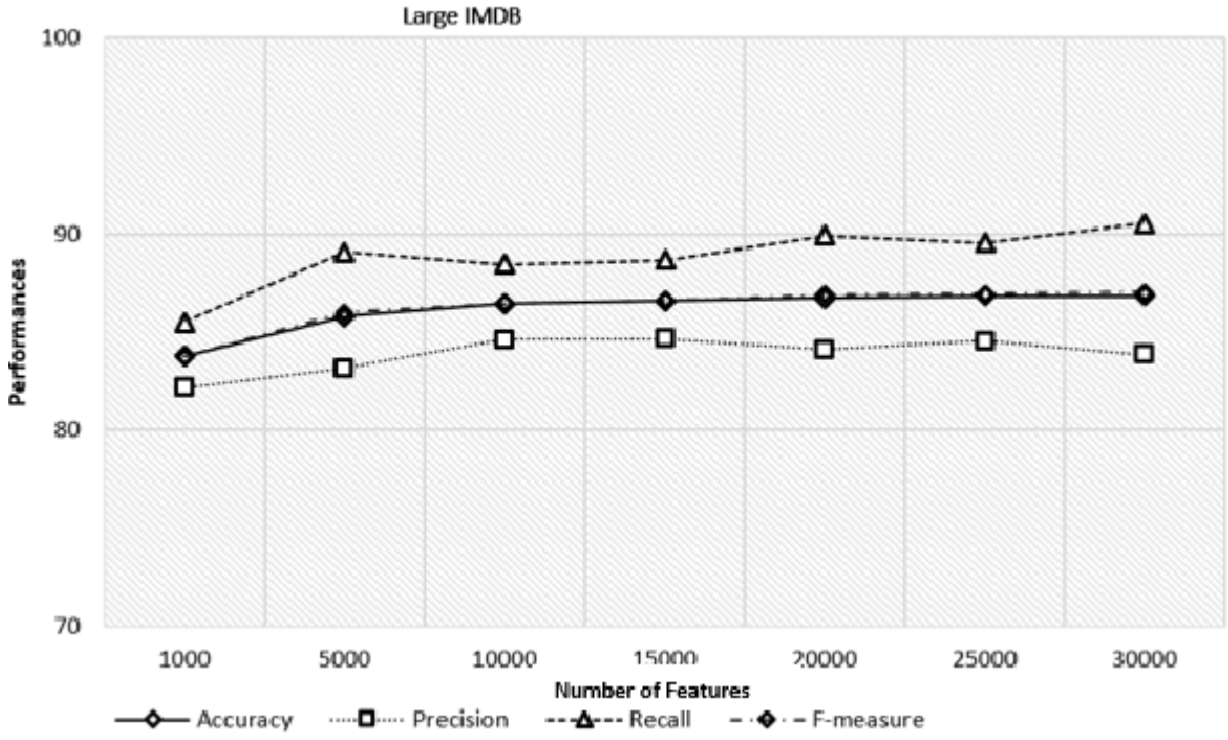

FIGURE 8. The performance of large IMDB dataset.

TABLE 13. Comparison of Performance Measurement from the First and Second Experiment.

Dataset		Without Feature Selection Technique					Proposed Technique				
		Num. of Features	Accuracy	Precision	Recall	F-measure	*k*-top Features	Accuracy	Precision	Recall	F-measure
Sentiment Labelled	IMDB	2039	82.66	80.98	83.75	82.34	2000	82.31	80.11	84.27	82.13
	Amazon	1174	78.65	77.24	83.11	80.04	700	80.05	84.06	75.63	79.61
	Yelp	1355	84.14	87.41	83.78	85.55	400	84.04	88.03	82.79	85.33
Large IMDB		30959	83.16	82.24	83.70	82.96	25000	86.85	84.54	89.56	86.96

TABLE 14. Comparison of Accuracy Performance with Other Techniques.

Dataset	Technique				
	Madasu&Elango[19]	Larasati *et al.*[5]	Bhuvaneswari&Parimala[24]	Al Amrani *et al.*[18]	Proposed Technique
IMDB	76.67	80.2	-	-	82.31
Amazon	-	-	87.5	83.4	80.05
Yelp	81.33	-	83.5	-	84.04

& Elango and 83.5% for Bhuvaneswari & Parimala. It is 0.54% and 2.71% higher than state-of-the-art techniques. For classification, the proposed technique obtained 400 features out of 1355 features. The most recent studies, however, do not specify the number of features obtained.

Note that, Madasu & Elango implemented TF-IDF as feature extraction and tested it with various feature selections and the classifier [19]. Whereas Larasati *et al.* implemented a hybrid feature selection technique which is the combination of TF-IDF and Chi-square and used SVM as the classifier. [5]. Bhuvaneswari & Parimala implemented a hybrid feature selection which is 'Synsets' feature set coupled with feature correlation, and SVM as the classifier [24].

However, when the Amazon category is tested, the accuracy of our proposed technique is lower than that of the other techniques. While compared to our proposed technique, the highest accuracy is 87.5 percent from Bhuvaneswari and Parimala, and 83.4% from Al Amrani *et al.* Al Amrani *et al.* implement RFSVM, a hybridization of Random Forest and SVM. The SVM classifier is employed [18].

Overall, the proposed feature selection technique gives a competitive score in terms of the accuracy especially in IMDB and Yelp category. Unfortunately, the state-of-art techniques did not provide the number of selected features in their paper after the implementation of the respective feature selection technique. Thus, we cannot compare the efficiency in terms of feature reduction during the classification.

V. SUMMARY AND CONCLUSION

For sentiment classification, this paper proposes an enhanced hybrid feature selection technique. TF-IDF + SVM-RFE is a technique that was designed, developed, and tested to improve the performance of the existing ML-based feature selection technique. Two datasets for sentiment analysis are retrieved. We pre-processed the datasets in order to extract important features. The proposed feature selection technique was then developed and tested. In the classification process, an SVM classifier is used.

Finally, the proposed feature selection technique successfully improved the classification performances in two datasets: Sentiment Labelled (Amazon category) and large IMDB dataset. It meets the human baseline agreement range (80% - 85% agreement) and is capable of exceeding the range in certain performance measures. This technique has also demonstrated the ability to reduce feature size from 19.25% to 70.50% while maintaining optimal classification performance. Simultaneously, no language-specific dictionary is used during the text document's pre-processing. The proposed feature selection does not improve classification performance in the other two datasets: Sentiment Labelled IMDB and Yelp dataset. These are most likely due to limited number of high-quality training data.

Based on this research, we successfully design, develop, and test an enhanced hybrid feature selection technique that combines TF-IDF and SVM-RFE to address limitations in filter and wrapper feature selection approaches as well as issues with existing hybrid feature selection techniques. We examine our proposed technique, which outperforms state-of-the-art technique in terms of accuracy. It is a good sign because we applied the proposed feature selection technique to a variety of sentiment review datasets, which will be useful in a variety of industries. However, a technical limitation emerged during the development of the proposed technique, preventing us from testing it on larger datasets.

As for future work, the result of this research motivates us the following future directions. First, the proposed feature selection technique will be tested on a larger dataset to measure its efficiency and effectiveness. Another research direction will focus on determining the best threshold in selecting features from the TF-IDF feature list.

REFERENCES

[1] W. Wang, H. Wang, and Y. Song, "Ranking product aspects through sentiment analysis of online reviews," *J. Exp. Theor. Artif. Intell.*, vol. 29, no. 2, pp. 227–246, Mar. 2017, doi: 10.1080/0952813X.2015.1132270.

[2] H. Nguyen, A. Veluchamy, M. Diop, and R. Iqbal, "Comparative study of sentiment analysis with product reviews using machine learning and lexicon-based approaches," *SMU Data Sci. Rev.*, vol. 1, no. 4, p. 7, 2018.

[3] W. AL-Saiagh, S. Tiun, A. AL-Saffar, S. Awang, and A. S. Al-khaleefa, "Word sense disambiguation using hybrid swarm intelligence approach," *PLoS ONE*, vol. 13, no. 12, Dec. 2018, Art. no. e0208695.

[4] T. Sabbah, M. Ayyash, and M. Ashraf, "Hybrid support vector machine based feature selection method for text classification," *Int. Arab J. Inf. Technol.*, vol. 15, no. 3, pp. 599–609, 2018.

[5] U. I. Larasati, M. A. Muslim, R. Arifudin, and A. Alamsyah, "Improve the accuracy of support vector machine using chi square statistic and term frequency inverse document frequency on movie review sentiment analysis," *Sci. J. Informat.*, vol. 6, no. 1, pp. 138–149, May 2019, doi: 10.15294/sji.v6i1.14244.

[6] B. Das and S. Chakraborty, "An improved text sentiment classification model using TF-IDF and next word negation," 2018, *arXiv:1806.06407*. [Online]. Available: http://arxiv.org/abs/1806.06407

[7] M. Luo and L. Luo, "Feature selection for text classification using OR+SVM-RFE," in *Proc. Chin. Control Decis. Conf.*, Xuzhou, China, May 2010, pp. 257–274, doi: 10.1109/ccdc.2010.5498331.

[8] M. Ahmad, S. Aftab, M. Salman, and N. Hameed, "Sentiment analysis using SVM: A systematic literature review," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 2, pp. 182–188, 2018, doi: 10.14569/IJACSA.2018.090226.

[9] Z. Drus and H. Khalid, "Sentiment analysis in social media and its application: Systematic literature review," *Procedia Comput. Sci.*, vol. 161, pp. 707–714, 2019, doi: 10.1016/j.procs.2019.11.174.

[10] E. A. Hameed, F. Tahir, and M. A. Shahzad, "Empirical comparison of sentiment analysis techniques for social media," *Int. J. Adv. Appl. Sci.*, vol. 5, no. 4, pp. 115–123, Apr. 2018.

[11] S. S. Kumar and A. Rajini, "Extensive survey on feature extraction and feature selection techniques for sentiment classification in social media," in *Proc. 10th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2019, pp. 6–11.

[12] G. Tripathi and S. Naganna, "Feature selection and classification approach for sentiment analysis," *Mach. Learn. Appl., Int. J.*, vol. 2, no. 2, pp. 1–16, Jun. 2015, doi: 10.5121/mlaij.2015.2201.

[13] T. P. Sahu and S. Ahuja, "Sentiment analysis of movie reviews: A study on feature selection & classification algorithms," in *Proc. Int. Conf. Microelectron., Comput. Commun. (MicroCom)*, Durgapur, India, Jan. 2016, pp. 1–6.

[14] M. Avinash and E. Sivasankar, "A study of feature extraction techniques for sentiment analysis," in *Emerging Technologies in Data Mining and Information Security* (Advances in Intelligent Systems and Computing), vol. 814, A. Abraham, P. Dutta, J. Mandal, A. Dutta, and S. Bhattacharya, Eds. Singapore: Springer, 2019.

[15] A. Al-Saffar, S. Awang, H. Tao, N. Omar, W. Al-Saiagh, and M. Al-Bared, "Malay sentiment analysis based on combined classification approaches and Senti-lexicon algorithm," *PLoS ONE*, vol. 13, no. 4, 2018, Art. no. e0194852.

[16] Y. Al Amrani, M. Lazaar, and K. E. El Kadiri, "Random forest and support vector machine based hybrid approach to sentiment analysis," *Procedia Comput. Sci.*, vol. 127, pp. 511–520, 2018.

[17] A. Madasu and S. Elango, "Efficient feature selection techniques for sentiment analysis," *Multimedia Tools Appl.*, vol. 79, nos. 9–10, pp. 6313–6335, Mar. 2020, doi: 10.1007/s11042-019-08409-z.

[18] A. S. Ghareb, A. A. Bakar, and A. R. Hamdan, "Hybrid feature selection based on enhanced genetic algorithm for text categorization," *Expert Syst. Appl.*, vol. 49, pp. 31–47, May 2016, doi: 10.1016/j.eswa.2015.12.004.

[19] A. K. Uysal and S. Gunal, "A novel probabilistic feature selection method for text classification," *Knowl.-Based Syst.*, vol. 36, pp. 226–235, Dec. 2012, doi: 10.1016/j.knosys.2012.06.005.

[20] E. Montañés, J. R. Quevedo, E. F. Combarro, I. Díaz, and J. Ranilla, "A hybrid feature selection method for text categorization," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 15, no. 2, pp. 133–151, Apr. 2007, doi: 10.1142/S0218488507004492.

[21] N. K. Suchetha, A. Nikhil, and P. Hrudya, "Comparing the wrapper feature selection evaluators on Twitter sentiment classification," in *Proc. Int. Conf. Comput. Intell. Data Sci. (ICCIDS)*, Chennai, India, Feb. 2019, pp. 1–6, doi: 10.1109/ICCIDS.2019.8862033.

[22] K. Bhuvaneswari and R. Parimala, "Sentiment reviews classification using hybrid feature selection," *Int. J. Database Theory Appl.*, vol. 10, no. 7, pp. 1–12, Jul. 2017, doi: 10.14257/ijdta.2017.10.7.01.

[23] L. Dey, S. Chakraborty, A. Biswas, B. Bose, and S. Tiwari, "Sentiment analysis of review datasets using Naïve Bayes' and K-NN classifier," *Int. J. Inf. Eng. Electron. Bus.*, vol. 8, no. 4, pp. 54–62, 2016, doi: 10.5815/ijieeb.2016.04.07.

[24] F. Iqbal, J. M. Hashmi, B. C. M. Fung, R. Batool, A. M. Khattak, S. Aleem, and P. C. K. Hung, "A hybrid framework for sentiment analysis using genetic algorithm based feature reduction," *IEEE Access*, vol. 7, pp. 14637–14652, 2019, doi: 10.1109/ACCESS.2019.2892852.

[25] C. Buckley, "Implementation of the SMART information retrieval system," Cornell Univ., Ithaca, NY, USA, Tech. Rep., 1985.

[26] M. F. Porter, "An algorithm for suffix stripping," *Program, Electron. Library Inf. Syst.*, vol. 14, no. 3, pp. 313–316, 1980.
[27] L. H. Patil and M. Atique, "A novel approach for feature selection method TF-IDF in document clustering," in *Proc. 3rd IEEE Int. Advance Comput. Conf. (IACC)*, Ghaziabad, India, Feb. 2013, pp. 858–862.
[28] M. Goudjil, M. Koudil, M. Bedda, and N. Ghoggali, "A novel active learning method using SVM for text classification," *Int. J. Autom. Comput.*, vol. 15, no. 3, pp. 290–298, Jun. 2018, doi: 10.1007/s11633-015-0912-z.
[29] A. W. Haryanto and E. K. Mawardi, "Influence of word normalization and chi-squared feature selection on support vector machine (SVM) text classification," in *Proc. Int. Seminar Appl. Technol. Inf. Commun.*, Semarang, Indonesia, Sep. 2018, pp. 229–233.
[30] Y. Lin and J. Wang, "Research on text classification based on SVM-KNN," in *Proc. IEEE 5th Int. Conf. Softw. Eng. Service Sci.*, Beijing, China, Jun. 2014, pp. 842–844.
[31] H. Arafat, R. M. Elawady, S. Barakat, and N. M. Elrashidy, "Different feature selection for sentiment classification," *Int. J. Inf. Sci. Intell. Syst.*, vol. 3, no. 1, pp. 137–150, 2014.
[32] S. Aman and S. Szpakowicz, "Identifying expressions of emotion in text," in *Text, Speech and Dialogue* (Lecture Notes in Computer Science), vol. 4629, V. Matoušek and P. Mautner, Eds. Berlin, Germany: Springer, 2007.

NUR SYAFIQAH MOHD NAFIS received the master's degree in computer science from Universiti Sultan Zainal Abidin, Malaysia, in 2016. She is currently pursuing the Ph.D. degree with Universiti Malaysia Pahang, Malaysia. Her research interest includes data classification using machine learning.

SURYANTI AWANG received the Ph.D. degree in electrical engineering from Universiti Teknologi Malaysia, Johor Bahru, Malaysia, in 2014. She worked as a Research Officer with the Centre of Artificial Intelligence and Robotic (CAIRO), Universiti Teknologi Malaysia, from 2002 to 2005. She has been a Senior Lecturer (equivalent to an Assistant Professor) with the Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, Malaysia, since 2005 until now. She is the coauthor for more than 30 journals and conference papers. Her research interests include pattern recognition, machine learning, and soft computing. She has collaborating with many industries in developing artificial intelligence systems. She receives numerous research grants from agencies, including a grant from Ministry of Higher Education of Malaysia under the Fundamental Research Grant Scheme with title "A New Feature Selection Technique for Text Classification." She had been awarded with Gold Medal in MTE'19 for Vehicle Type Recognition System, Silver Award in MTE'18, ITEX'18, and ITEX'17, for other projects.