JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY
https://e-journal.uum.edu.my/index.php/jict

How to cite this article:
Mat Zin, H., Mustapha, N., Azmi Murad, M. A., & Mohd Sharef, N. (2022). A meta-heuristic algorithm for the minimal high-quality feature extraction of online reviews. *Journal of Information and Communication Technology, 21*(4), 571-593. https://doi.org/10.32890/jict2022.21.4.5

A Meta-heuristic Algorithm for the Minimal High-Quality Feature Extraction of Online Reviews

*[1]Harnani Mat Zin, [2]Norwati Mustapha,
[3]Masrah Azrifah Azmi Murad,
& [4]Nurfadhlina Mohd Sharef
[1]Computing Department, Faculty of Computing, Arts & Creative Industry,
Universiti Pendidikan Sultan Idris, Malaysia
[2,3&4]Department of Computer Science,
Faculty of Computer Science and Information Technology,
Universiti Putra Malaysia, Malaysia

*harnani@fskik.upsi.edu.my;
norwati@fsktm.upsi.edu.my;
masrah@fsktm.upm.edu.my;
nurfadhlina@fsktm.upm.edu.my
*Corresponding author

Received: 30/1/2022 Revised: 5/7/2022 Accepted: 6/7/2022 Published:20/10/2022

ABSTRACT

Feature extraction and selection are critical in sentiment analysis (SA) to extract and select only the appropriate features by removing those deemed redundant. As such, the successful implementation of this process leads to better classification accuracy. Inevitably, selecting high-quality minimal features can be challenging given the inherent complication in dealing with over-fitting issues. Most of the current studies used a heuristic method to perform the classification process that will result in selecting and examining only a single feature subset, while ignoring the other subsets that might give better results. This study explored the effect of using the meta-heuristic method together with the ensemble classification method in the sentiment classification of online reviews. Adding to that point, the extraction and selection of relevant features used feature ranking, hyper-parameter optimization, crossover, and mutation, while the classification process utilized the ensemble classifier. The proposed method was tested on the polarity movie review dataset v2.0 and product review dataset (books, electronics, kitchen, and music). The test results indicated that the proposed method significantly improved the classification results by 94%, which far exceeded the existing method. Therefore, the proposed feature extraction and selection method can help in improving the performance of SA in online reviews and, at the same time, reduce the number of extracted features.

Keywords: Feature extraction, feature selection, online reviews, meta-heuristics, sentiment analysis.

INTRODUCTION

Over recent years, advancements in Internet technology have developed rapidly, making it easier for users to convey their views of a wide spectrum of products, events, and services via online platforms, resulting in the exponential growth of online content. Admittedly, analyzing online content manually can be both tedious and backbreaking. Herein lies the need to automate the process of analyzing online sentiments of users. Essentially, sentiment refers to a user's view, feeling, or opinion of a product, event, or service that is posted over the Internet (Pang & Lee, 2008). Of late, sentiment analysis (SA) research has become popular, focusing on automating processes through which analysts can identify and extract opinions, attitudes, and sentiments from online content. Specifically, SA is a technique that analyzes users' opinions, moods, evaluations, judgments, attitudes, and feelings toward entities, such as products, services, organizations, people, issues, events, topics, and their attributes (Liu, 2012). Operationally, SA works by classifying texts into subjective or objective classes to help recognize positive or negative opinions regarding subjective texts (Al-Harbi, 2019). SA involves two main tasks, namely the extraction of features from online content and the classification of sentiments, which can be classified into negative and positive classes (Ekbal & Saha, 2013).

Currently, SA has replaced web-based surveys conducted by companies to gauge public opinion about products, events, or services (Asghar et al., 2014). SA also helps organizations examine users' perceptions of their products, events, or services, all of which are vital information that can help improve their decisions. Likewise, SA assists end-users in decision-making to choose a product, event, or service that they are interested in (Zin et al., 2018).

There are currently two main groups in the SA method: the machine learning approach and the lexicon-based approach (Pang et al., 2002). The former includes Support Vector Machine (SVM), Naïve Bayes, Maximum Entropy, Decision Tree, Random Forest, and Linear Discriminant Function (LDF), while the latter includes WordNet, SentiWordNet, SenticNet, and Multi-Perspective Question Answering (MPQA) that rely on the sentiment lexicon to determine the polarity of textual content. Interestingly, some studies have combined the above two approaches to building a lexicon-based classifier, which is called the hybrid approach (Behera & Roy, 2016). Equally fascinating, some recent studies of SA (published in 2018, 2019, and 2020) have introduced the deep learning approach. This deep learning approach is adopted from the machine learning approach, with Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) being the most popular deep learning techniques (Ligthart et al., 2021).

In any SA-related work, several processes are involved before generating the final results as follows: (1) the reading of reviews from a database, (2) the preprocessing of reviews to highlight any irrelevant words, (3) the feature extraction of relevant features, (4) the feature selection of essential features and removal of irrelevant or redundant features, and (5) the classification of group features according to their classes (Birjali et al., 2021). As such, each process in SA plays a vital role in achieving accurate classifications of good sentiments.

The feature extraction process identifies the features or aspects of a product, event, or service that reviewers have commented on (Asghar et al., 2014). In feature extraction, texts are used as input from which relevant features are extracted using various techniques, such as n-gram, part-of-speech (POS) tags, function words, and word-based features (Shahana & Omman, 2015). Some important features in SA are the presence and frequency of terms, POS tags, opinion words and phrases, and negations (Birjali et al., 2021).

In principle, feature selection is a process that reduces the dimensionality of a feature space by identifying and choosing relevant features and removing unnecessary, irrelevant, or redundant features. Ultimately, the principal objective of feature selection is to reduce the dimensionality of feature space and the over-fitting of the learning scheme of training data (Kummer & Savoy, 2012). Depending on their specific objectives, feature extraction and selection techniques can be categorized as follows: (1) techniques to overcome the problem of over-fitting and improve the classification performance, (2) techniques to reduce processing time and improve cost efficiency, and (3) techniques to understand the basic process of generating data (Yousefpour et al., 2017).

As highlighted, there are two main feature selection techniques, namely the lexicon-based and statistical methods. The former is effective for interpreting and extracting features that humans manually create. However, such an approach is difficult to perform as it takes considerable time to select such created features (Duric & Song, 2012). SentiWordNet8 is a popular example of this approach (Baccianella et al., 2014). On the other hand, the statistical method is based on a statistical approach, which is fully automatic and widely used for feature selection. Nevertheless, such an approach often fails to separate relevant features from redundant ones (Duric & Song, 2012). Essentially, the statistical approach consists of four categories: filter, wrapper, embedded, and hybrid (Hoque et al., 2014). No learning algorithms are used in the filter approach to select a feature subset. In contrast, learning algorithms are used in the wrapper approach to evaluate accuracy. Likewise, learning algorithms are employed in the embedded approach to select relevant features during the training process. Interestingly, the hybrid approach is based on the combination of filter and wrapper-based approaches.

The sentiment classification task aims to build a more accurate classification model based on training samples from reviews. However, sentiment classification is fraught with problems attributed to huge dimensions and unwanted or redundant features. Notably, identifying high-quality features is a prevalent problem in sentiment classification based on the machine learning method. In this regard, feature selection is one of the critical processes to overcome this problem by selecting the optimal feature subset from a feature list, which is evaluated based on certain criteria the researcher has set. Another problem besetting this classification is over-fitting, which can occur when a classifier is over-trained. Most of the current studies used the heuristic approach to extract and select the features for the classification process. This method only selects and examines a single feature subset and ignores the other subsets that might give better results.

In this study, the researchers propose a feature selection method to extract a high-quality minimal subset of features from a real-world setting. The proposed method is based on a hybrid filter and a wrapper approach to reduce the dimensionality of a high-dimensional feature subset space. In particular, the wrapper approach uses a hybrid of heuristic and meta-heuristic strategies to generate a subset of features involving several steps. First, the heuristic strategy is utilized to determine the initial feature subsets. Then, a differential evaluation method, which belongs to the genetic class of meta-heuristic strategies, is employed to improve the initial feature subsets by applying the meta-heuristic search. The ensuing discussions of this paper are organized into several sections. Section 2 discusses the related works of SA, while Section 3 elaborates on the proposed method. Then, Section 4 analyzes the results of the comparative experiments and evaluations carried out in this study, followed by Section 5, which concludes the discussions of the paper.

RELATED WORKS

In SA, feature selection is crucial to reduce the complexity of the classification process and eliminate the issue of over-fitting (Rajpoot et al., 2021). In the literature, feature selection is also referred to as attribute selection, variable selection, or variable subset selection (Kaur, 2017). According to Kaur (2017), feature selection is a process of identifying and choosing a subset of relevant features that will later be used in model construction. There are three main objectives of feature selection: (1) to enhance the performance of a classifier, (2) to provide efficient, inexpensive predictors, and (3) to provide a better understanding of the essential process that generates data (Iguyon & Elisseeff, 2003).

Depending on the input data type, which can be either labeled or unlabeled, feature selection algorithms can be divided into supervised, unsupervised, and semi-supervised feature selections (Tang et al., 2014). In turn, supervised feature selection can be divided into three methods, i.e., filter, wrapper, and embedded models. The first method provides a means to select an optimal subset of features by determining a scoring function, such as selecting and eliminating high-scoring and low-scoring features. For the second method, the selection of an optimal feature subset is performed by generating and evaluating different subsets in a feature subset space and extracting them using a classifier. For the third method, the search is performed by combining a model hypothesis and a feature subset space in a classifier structure. Table 1 illustrates the differences between these three methods.

Table 1

The Strengths, Weaknesses, and Examples of Feature Selection Methods (Yousefpour, Ibrahim, Nuzly, & Hamed (2014a); Naheed, Shaheen, Khan, Alawairdhi, & Khan (2020))

Type		Strengths	Weaknesses	Examples of Technique
Filter method	Univariate	- Quick - Gradable - Non-dependence of classifiers	- Relinquished dependence on features - Relinquished interplay with classifiers	- Information Gain (IG) - Chi-square (CHI) - T-test
	Multivariate	- Dependence of features - Independence of classifiers - Better time complexity than wrapper	- Slower than univariate methods - Less gradable than univariate methods - Relinquished interplay with classifier	- Correlation-based feature selection (CFS) - Markov blanket filter (MBF) - Fast correlation-based feature selection (FCBF)

(continued)

Type		Strengths	Weaknesses	Examples of Technique
Wrapper method	Univariate	- Simple - Dependence of features - Interact with the classifier - Slower than randomize	- High risk of over-fitting - More chance of entrapment with local optimum than randomize - Classifier-dependent selection	- Sequential forward selection (SFS) - Sequential backward elimination (SBE) - Beam search
	Multivariate	- Dependence of features - Less entrapment with local optimum - Interplay with classifiers	- Classifier-dependent selection - Greater risk of over-fitting than the deterministic method	- Simulated annealing - Genetic algorithm
Embedded method		- Dependence of features - Interplay with classifiers - Better time complexity than wrapper	- Classifier-dependent selection	- Decision Tree - Weighted Naïve Bayes - Feature selection using the weight vector of SVM

In recent years, many studies have focused on the feature selection method. For example, Novaković, Strbac, and Bulatović (2011) introduced Information Gain (IG), Gain Ratio (GR), Symmetrical Uncertainty (SU), Relief-F (RF), One-R (OR), and chi-square statistic (CHI) as feature selection methods. Likewise, a study by Tang et al. (2014) involved IG, Mutual Information (MI), CHI, GR, and Document Frequency (DF). Later, Yousefpour et al. (2014a) introduced a feature selection method using standard deviation and compared its results with those of IG and CHI. The results showed that the former was more accurate than the latter. In other related works, Minimum Redundancy Maximum Relevance (mRMR) was used as a feature selection method that achieved better performance than IG (Agarwal & Mittal, 2016). Meanwhile, Manek et al. (2016) and Kaur (2017) used Gini index as a feature selection technique, which led to better results. Meanwhile, some recent studies have applied deep learning methods to perform feature selection, such as using Query Expansion Ranking (QER) (Parlar et al., 2018) and Modified Categorical Proportional Difference (MCPD) (Chang et al., 2020). Better classification performance is observed when deep learning is applied.

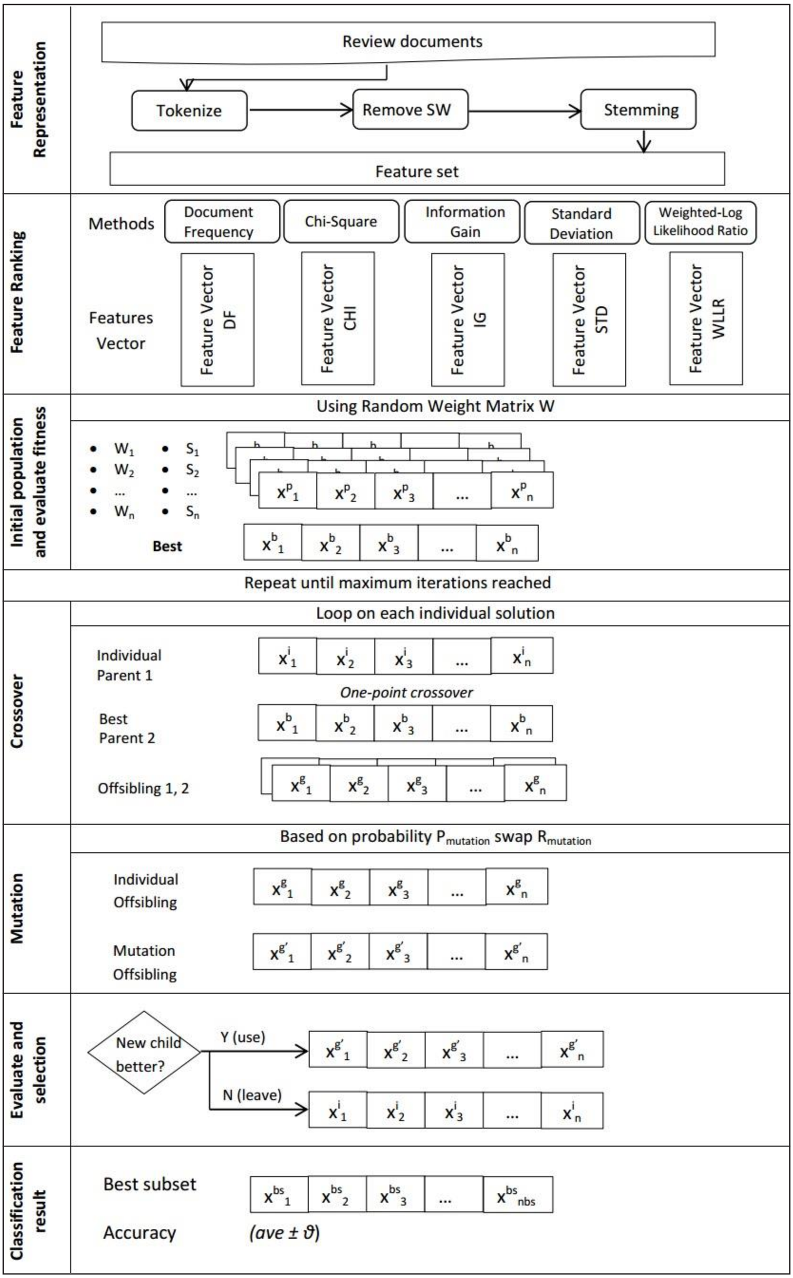
As highlighted, over-fitting is one of the major problems affecting SA. In recent years, several studies have been conducted that used the k-fold cross-validation technique to solve over-fitting (Wijesinghe, 2017; Diwakar, 2019). Similarly, Manalu (2020) attempted to solve such a problem using an early stopping function. In view of these recent works, a new method was proposed using a combination of both techniques, namely k-fold cross-validation and early stop, together with the ensemble classifier method. An ensemble classifier method is an approach that applies several single classifiers where the classification will be identified for classifying new unseen features (Sainin et al., 2021). The proposed method was tested through several experiments that produced some promising results.

THE PROPOSED META-HEURISTIC METHOD FOR FEATURE SELECTION

This section describes the proposed method used in this study that involved machine learning algorithms to divide reviews into positive or negative classes. This study was performed through seven main phases, as shown in Figure 1: feature representation, feature ranking, initial population and fitness evaluation, crossover, mutation, evaluation and selection, and classification.

Figure 1

The Proposed Meta-Heuristic Method for Feature Selection



Dataset

This study used five document-level datasets consisting of movie, book, electronics, kitchen, and music reviews, which have been widely used in many SA studies. Specifically, these datasets were applied to evaluate the performance of the proposed method. All the datasets were annotated at the document level, and only two polarity classes were considered in this study, namely positive and negative classes. Table 2 highlights the statistical descriptions of the datasets, the details of which are described as follows:

- The datasets used by Pang and Lee (2004) are a set of movie review documents in terms of their overall sentiment polarity. It is freely available at http://www.cs.cornell.edu/people/pabo/movie-review-data/
- The datasets used by Blitzer, Dredze, and Pereira (2007) contain Amazon product reviews covering different product types belonging to 25 different domains, such as book reviews, electronics reviews, and music reviews. This dataset can be accessed at http://www.cs.jhu.edu/ ~ mdredze/datasets/sentiment/

Table 2

The Statistical Description of Datasets

Dataset	# positive sample	# negative sample	# features in positive sample	# features in negative sample	# total features
Movie	1,000	1,000	35,492	33,184	48,690
Book	1,000	1,000	15,995	15,607	23,934
Electronics	1,000	1,000	8,063	7,864	11,594
Kitchen	1,000	1,000	7,150	6,863	10,249
Music	1,000	1,000	13,101	12,224	19,404

Feature Representation

This first phase comprised the preprocessing step and the feature representation process that used unigram-based features as the feature representation. In this phase, the detection of features from the raw documents involved three stages: (1) the tokenization stage to scan texts and identify words, (2) the stop-word removal stage to delete noises and meaningless words, and (3) the stemming stage to reduce inflected or derived words.

Feature Ranking

In this phase, the extracted features were evaluated using various ranking methods, such as DF, IG, CHI, standard deviation (STD), and weighted-log likelihood ratio (WLLR). The following formula represents each set of features extracted in the first phase: $F = \{f_1, f_2, \ldots, f_N\}$, where N is the total number of features and f_i signifies a feature. Each feature could be ranked by several feature ranking methods, namely $M_1, M_2, \ldots, M_L$. The features were first weighted by a feature ranking method (DF, IG, STD, and WLLR) and then sorted in descending order according to their weights to create a feature vector (FV). A feature vector $FV_j = [f_{i1}^j, f_{i2}^j, \ldots, f_{iN}^j]$ created by the j^{th} feature ranking method was considered as a permutation of F. Here, f_{i1}^j could be represented as x_i, such that $FV = [x_1, x_2, \ldots, x_N]$, where x_1 had the highest rank (or weight) and x_2 had the second-highest rank among the feature vectors, as shown in Equation 1.

$$F = \{f_1, f_2, \ldots, f_N\} \rightarrow FV = [x_1, x_2, \ldots, x_n] \qquad (1)$$

Initial Population and Fitness Evaluation

In this phase, the initial population matrix $S = \{s_{1,} s_{2,} \ldots, s_{NP}\}$ was created. The matrix size is N x NP, where N is the number of features and NP is the number of population. This phase was performed in two stages in which a weight matrix was used to create the required solution by integrating the FV developed in the previous phase. The following formula represents the creation of the initial population ranking method weight: $W = \{w_{1,} w_{2,} \ldots, w_{NP}\}$ with a matrix size of L x NP, where L is the number of ranking methods and NP is the number of population. In the second stage, the weight matrix was utilized to create the initial population matrix S by ranking the features by the sum of the weights of different ranking methods multiplied by the weight of the ranking method created by the initial population ranking method weights W. Then, the features were ranked and arranged from top to bottom in terms of their weights based on Equation 2 and using Algorithm 1.

$$Rank\ (f_i^s) = \mathring{a}_{l=1}^{L}\ W_{ls} * Index(f_t^l) \tag{2}$$

Where index x is the place of f_i in FV ranked by ranking method L

The evaluation was carried out on each solution for the population matrix to achieve the highest accuracy and determine the corresponding feature subset for each solution. Essentially, this phase consisted of four stages. In the first stage, each solution for the population subset was generated according to Equation 3. In the second stage, the sample reviews were split into the 5-fold cross-validation with a random start to avoid over-fitting problems. Several training sets were generated from all permutations for the four folds, with the remaining fold considered the test set. Overall, five training sets were generated in this second stage.

In the third stage, various classification algorithms were used to build a machine learning model based on the training sets generated in the previous stage using the subsets generated in the first stage. Hyperparameter optimization was also used to improve the results of the learning process. Then, the classification score for each classification algorithm was normalized using Equation 4 and subsequentl, summed up to avoid over-fitting problems. Equation 5 was used to obtain the class of each document for testing each feature subset generated in the first stage. Furthermore, training was stopped early to avoid over-fitting problems. If the test accuracy started to decrease while the training accuracy was still increasing for the number of iterations that equaled the number of $N_{\text{early stop}}$, the classifier was forced to stop. Therefore, the highest accuracy was attained, and a corresponding subset for five receptions was selected in this fourth stage.

$$\begin{gathered} S_i = \{x_1, x_2, x_3, \ldots, x_n\} \\ Feature\ subsets = \{\{x_1\}, \{x_1, x_2\}, \{x_1, x_2, x_3\}, \ldots, \{x_1, x_2, x_3, \ldots. x_n\}\} \end{gathered} \tag{3}$$

$$Classification\_Score = \frac{Classification\ Score - Min\ Classification\ Score}{Max\ Classification\ Score - Min\ Classification\ Score} \tag{4}$$

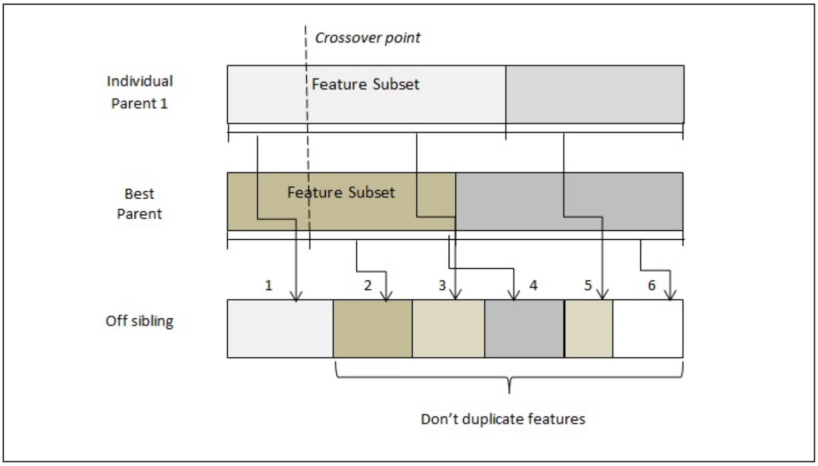
$$Classification(d_t) = \begin{cases} \dfrac{\sum_{c=1}^{C} Classification\ Score\ C(d_t)}{C} \geq .5\ Positive \\ \dfrac{\sum_{c=1}^{C} Classification\ Score\ C(d_t)}{C} < .5\ Negative \end{cases} \tag{5}$$

Crossover

In this phase, the crossover operation between the best solution (based on the accuracy obtained in the previous phase) and each solution for the population was performed involving Algorithm 2. Specifically, the one-point crossover was used when a point was less than the minimum value of the two solutions. A sibling solution was generated by adding the first part of the best solution to the first part of the population solution. Then, the second part of the best solution was added to the second part of the population solution. Repeated features in the sibling solution were removed, starting with the first feature. Figure 2 depicts the operations of the crossover performed in this study.

Figure 2

The Crossover Operation

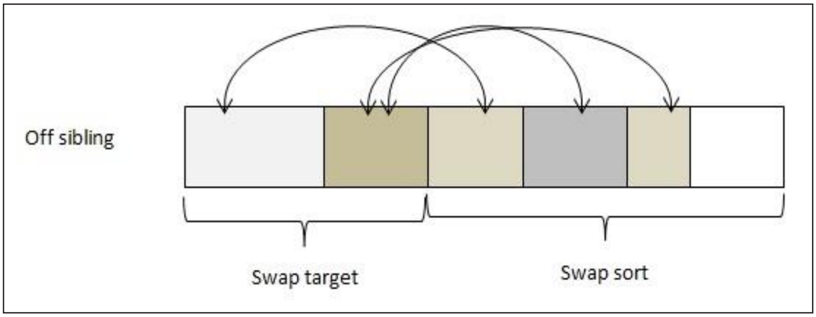


Mutation

The execution of this phase was based on the $P_{Mutation}$ probability for each sibling generated in the previous phase. In this phase, a ratio of features defined by $R_{Mutation}$ was randomly selected before the crossover points and moved to random locations after the crossover point. Figure 3 shows the mutation process performed in this phase.

Figure 3

The Mutation Operation



Evaluation and Selection

In this phase, the fitness of a new sibling was evaluated. Based on the evaluation results, the new sibling would be included in the population if its accuracy was higher than that of one of its parent solutions.

Classification Result

This final phase revealed the experimental results by highlighting the best solution after repeating the fourth to seventh steps based on the $\text{Iteration}_{\text{Max}}$ times.

Algorithm 1: Generate Solution FV

```
Input:
F: extracted features vector
w: ranking methods weights

Output:
Feature Vector (FV)

Rank the FV as follows:
For i = 1: number of ranking methods
Apply i^th feature ranking method on training set
        Create i^th feature vector and sort it in descending order
End i

Rank features in F by weight w using Equation 2
Report FV
```

Algorithm 2: The Cross-Validation Algorithm for Evaluating each Solution Feature Vector

```
Input:
TDM based on TF-IDF weighting for solution feature
vector s with dimension N

Output:
Most relevant of features subsets generated by Equation 3
Average and standard deviation of performance measured for
most relevant features subset

For pass = 1: num_of_Repetitions
    Initialize first-fold on samples with a random start
    For fold = 1: num_K-fold
        Set current fold as test set and the remaining fold
        astraining sets
        Generate feature subsets incrementally based on
        Equation 3
        For wrap = 1: num_FeatureSubsets
            For classifier = 1: num_Classifiers
                Train classifier for current feature subsets
                for TDM
            End classifier
            Evaluate current feature subsets for TDM
            based on Equation 5
            If over-fitting condition occurs
                Exit Loop
            End
        End wrap
        Save feature subset with highest accuracy
        Adjust next fold
    End fold
End pass

Report most relevant feature subsets with average and standard
deviation of performance measures
```

Algorithm 3:
The Hybrid DE and Ordinal-based Algorithm for Feature Subset Selection

```
Input:
Document-level review documents
numMaxIterations: Number of maximum iterations
NP: Number of initial population
$P_{Mutation}$: Mutation probability
$R_{Mutation}$: Mutation percentage

Output:
Subset of most relevant features
Average and standard deviation of performance measured for
most relevant of features subset
Represent features as unigram-based
Create TDM for extracted features and weight based on TF-IDF
Create random ranking methods matrix W with dimension L x NP

For i = 1: NP
      Create initial population matrix individual i with $W_i$ using
      Algorithm 1
      Evaluate individual i using Algorithm 2
      Save individual i evaluation
End i

For t = 1: numMaxIterations
        Find best individual $S_{best}$

        For i = 1: NP
        Generate $S_s$ by crossover $S_{best}$ and $S_i$
              If rand < $P_{Mutation}$
                  Apply mutation for $S_s$ with percentage $R_{Mutation}$
              End
              Evaluate $S_s$ performance using Algorithm 2
              If $S_s$ performance > $S_i$ performance
                      Replace $S_i$ with $S_s$ in solutions matrix
              End
        End i
End numMaxIterations

Report most relevant feature subsets with average and standard
deviation of performance measures
```

RESULTS AND DISCUSSION

As highlighted, five review datasets (Pang & Lee, 2004; Blitzer et al., 2007) were used to examine the performance of the proposed technique. In particular, the 5-fold cross-validation was carried out to test its performance based on the above datasets by repeating a random starting point five times. In addition, three folds were used for training the classifier, one fold for hyperparameter optimization, and the last fold for testing. Table 3 shows the results of the analysis performed on the proposed method by highlighting the highest accuracy for each separate classifier based on the number of features used. Furthermore, the early stop point was added to each classifier, and the classifier was made to stop running when it met the stopping criteria to overcome over-fitting. As shown, the final training results using Equation 5 for all classifiers of each dataset were summarized in the last row.

Table 3

The Classification Accuracy, Number of Features, and Early Stop Length of Feature Subset of Unigram-based Features using Algorithm 3 method in 5∗5-FCV

Dataset	Classifier	Accuracy	# of Features	Early Stop # of Features
Movie	SVM	92.43 ± 0.74	10,236 ± 501	11,431 ± 124
	NB	91.27 ± 0.78	11,987 ± 731	12,506 ± 314
	ME	90.65 ± 0.93	4,342 ± 361	4,849 ± 123
	LDF	89.53 ± 0.72	5,887 ± 541	6,575 ± 184
	Ensemble	**92.92 ± 0.34**	**4,236 ± 501**	**5,146 ± 112**
Book	SVM	90.36 ± 0.89	6,821 ± 365	7,123 ± 431
	NB	89.21 ± 0.12	7,491 ± 136	7,791 ± 553
	ME	88.58 ± 0.82	2,892 ± 281	3,443 ± 531
	LDF	87.62 ± 0.91	3,913 ± 419	4,125 ± 846
	Ensemble	**91.68 ± 0.97**	**2,712 ± 124**	**3,124 ± 215**
Electronic	SVM	88.46 ± 0.92	5,123 ± 241	5,432 ± 211
	NB	89.58 ± 1.22	5,631 ± 859	5,814 ± 113
	ME	87.97 ± 0.31	2,281 ± 152	2,441 ± 231
	LDF	86.83 ± 0.73	2,981 ± 712	3,591 ± 291
	Ensemble	**90.66 ± 0.72**	**2,631 ± 859**	**2,814 ± 113**

(continued)

Dataset	Classifier	Accuracy	# of Features	Early Stop # of Features
Kitchen	SVM	91.21 ± 0.33	3,124 ± 295	4,012 ± 192
	NB	90.57 ± 0.89	4,215 ± 381	4,627 ± 261
	ME	90.03 ± 0.28	3,346 ± 381	4,182 ± 369
	LDF	91.24 ± 1.14	2,247 ± 113	2,413 ± 542
	Ensemble	**91.63 ± 0.54**	**2,173 ± 237**	**2,651 ± 327**
Music	SVM	89.12 ± 0.76	4,651 ± 274	5,128 ± 234
	NB	89.73 ± 1.21	6,176 ± 819	6,266 ± 261
	ME	88.01 ± 0.44	4,414 ± 491	5,261 ± 262
	LDF	90.87 ± 0.97	3,432 ± 143	4,152 ± 142
	Ensemble	**91.07 ± 0.81**	**3,213 ± 271**	**3,921 ± 229**

Table 4 demonstrates the comparison of the classification accuracy of the proposed method and that of a baseline work. The classification accuracy was evaluated based on the voting results for the test features of all classifiers using Equation 5 and Algorithm 3. The results were compared with the word-relation unigram features proposed by Xia et al. (2011) as the first baseline work, and ordinal-based feature vector (OIFV) and frequency-based feature vector (FIFV) feature integration proposed by Yousefpour et al. (2017) as the second baseline work.

Table 4

The Comparison of Results between the Proposed Method and Baseline Works

Dataset	Baseline 1: WR- based unigram feature (unigram)	Baseline 2: Integration-based feature		Proposed method: Meta-heuristic algorithm	
		OIFV	FIFV	5*5 FCV	Ensemble classifier
Movie	84.75	90.70	90.76	92.43	92.92
Book	74.70	84.72	85.13	90.36	91.68
Electronics	80.05	85.73	85.97	88.46	90.66
Kitchen	83.25	85.95	86.83	91.21	91.63
Music	77.20	84.57	85.64	89.12	91.07

Interestingly, the test revealed several results that may have to be given some careful consideration. First, the effectiveness of the proposed method was found to be highly promising. This study examined the effectiveness of ensemble classification in terms of accuracy and the number of features. As shown in Table 3, the effectiveness of the ensemble classification, as demonstrated in all experiments, far exceeded that of the method based on a single classifier. It is clear that, in addition to the classification score, the application of voting for each classifier based on the document classification yielded a higher score that provided greater confidence for the document polarity based on the classification result.

Second, the results could provide greater insight into how over-fitting could be avoided by the number of early stops needed to reduce classification time. As revealed, the classification time increased as the number of classification features increased, and any effort to stop the classification early resulted in a significant reduction in the classification time. Table 3 demonstrated the average number of features where Algorithm 3 met the early stop criteria to avoid over-fitting. The results showed that Algorithm 3 stopped executing when the average number of features equaled 1 percent of the maximum number of features, which reduced the execution time of Algorithm 3 due to the early stop.

Finally, the test results helped determine whether the proposed methods could outperform existing feature selection methods in classifying sentiment. The findings were achieved by comparing the current test results of the proposed method with those of the baseline works of Xia et al. (2011) and Yousefpour et al. (2017). As indicated in Table 4, the accuracy of the former was relatively higher than those of the latter, signifying that the proposed method significantly outperformed the methods used in previous works. Therefore, it is proven that the combination of the filter and wrapper approaches is able to reduce the dimensionality of feature subsets (as shown in Table 3), as well as increase the classification results (as demonstrated in Table 4).

CONCLUSION AND FUTURE RESEARCH

Extracting important, relevant features is the most exciting yet challenging task in the sentiment analysis of online reviews. Such a task is crucial to attaining better classification results. This study proposed a hybrid of the filter and wrapper methods with a meta-heuristic algorithm to determine a minimal high-quality subset of features extracted from a real domain. As demonstrated, the early stopping method helped eliminate over-fitting, which has been plaguing the classification process all these years. Furthermore, the ranking methods helped filter the essential features and reduce the dimensionality of feature space. Therefore, the proposed method could help increase the performance of SA work on online reviews and reduce the dimensionality of features.

Arguably, more studies are needed to test the proposed method with other large datasets, such as huge movie review datasets consisting of 50,000 review documents. Moreover, feature selection can also be extended to other n-gram features, such as bi-grams or tri-grams. It would also be interesting to combine such features with their semantic meanings by applying the semantic parser in future research. Overall, the meta-heuristic algorithm combined with the early stopping method can yield a higher classification performance of the extraction process of online documents or content.

ACKNOWLEDGMENT

The authors wish to express their gratitude to Sultan Idris Education University and University of Putra Malaysia for the support rendered in this research. The principal author also wishes to extend her deepest appreciation to the Ministry of Higher Education (MoHE) Malaysia for the scholarship that helped her pursue a doctoral study.

REFERENCES

Agarwal, B., & Mittal, N. (2016). Machine learning approach for sentiment analysis. In *Prominent feature extraction for sentiment analysis* (pp. 21–45). Springer, Cham.. https://doi.org/10.1007/978-3-319-25343-5_3

Al-Harbi, O. (2019). A comparative study of feature selection methods for dialectal Arabic sentiment classification using support vector machine. *International Journal of Computer Science and Network Security*, *19*(1), 167–176.

Asghar, M. Z., Khan, A., Ahmad, S., & Kundi, F. M. (2014). A review of feature extraction in sentiment analysis. *Journal of Basic and Applied Research International*, *4*(3), 181–186. https://doi.org/10.3233/IDA-173763

Baccianella, S., Esuli, A., & Sebastiani, F. (2014, May). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)*. European Language Resources Association (ELRA).

Behera, R. N., & Roy, M. (2016). Ensemble based hybrid machine learning approach for sentiment classification- A review. *International Journal of Computer Applications*, *146*(6), 31–36.

Birjali, M., Kasri, M., & Beni-Hssane, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, *226*, 107–134. https://doi.org/10.1016/j.knosys.2021.107134

Blitzer, J., Dredze, M., & Pereira, F. (2007, June). Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification John. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (pp. 440–447).

Chang, J.-R., Liang, H.-Y., Chen, L.-S., & Chang, C.-W. (2020). Novel feature selection approaches for improving the performance of sentiment classification. *Journal of Ambient Intelligence and Humanized Computing*, 1–14. https://doi.org/10.1007/s12652-020-02468-z

Diwakar, D. (2019, December). Proposed machine learning classifier algorithm for sentiment analysis. In *2019 Sixteenth International Conference on Wireless and Optical Communication Networks (WOCN)* (pp. 1–6). IEEE.

Duric, A., & Song, F. (2012). Feature selection for sentiment analysis based on content and syntax models. *Decision Support Systems*, *53*(4), 704–711. https://doi.org/10.1016/j.dss.2012.05.023

Ekbal, A., & Saha, S. (2013). Combining feature selection and classifier ensemble using a multiobjective simulated annealing approach: Application to named entity recognition. *Soft Computing*, *17*(1), 1–16. https://doi.org/10.1007/s00500-012-0885-6

Hoque, N., Bhattacharyya, D. K., & Kalita, J. K. (2014). MIFS-ND: A mutual information-based feature selection method. *Expert Systems with Applications*, *41*(14), 6371–6385.

Iguyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182. https://doi.org/10.1162/153244303322753616

Kaur, R. (2017). Sentiment analysis of movie reviews: A study of machine learning algorithms with various feature selection methods. *International Journal of Computer Sciences and Engineering,* *5*(9), 113–121. https://doi.org/10.26438/ijcse/v5i9.113121

Kummer, O., & Savoy, J. (2012, January). Feature selection in sentiment analysis. In *Conférence en Recherche d’Infomations et Applications (CORIA)* (pp. 273–284).

Ligthart, A., Catal, C., & Tekinerdogan, B. (2021). *Systematic reviews in sentiment analysis: A tertiary study. Artificial Intelligence Review*, *54*(7), 4997–5053. https://doi.org/10.1007/s10462-021-09973-3

Liu, B. (2012). Sentiment analysis and opinion mining. In *Synthesis lectures on human language technologies* (pp. 1–167). Morgan & Claypool Publishers. https://doi.org/10.1142/9789813100459_0007

Manalu, B. U., Tulus., & Efendi, S. (2020, September). Deep learning performance in sentiment analysis. In *2020 4th International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM)* (pp. 97–102). IEEE.

Manek, A. S., Shenoy, P. D., Mohan, M. C., & Venugopal, K. R. (2016). Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier. *World Wide Web*, *20*(2), 135–154. https://doi.org/10.1007/s11280-015-0381-x

Naheed, N., Shaheen, M., Khan, S. A., Alawairdhi, M., & Khan, M. A. (2020). Importance of features selection, attributes selection, challenges and future directions for medical imaging data: A review. *Computer Modeling in Engineeting & Sciences (CMES)*, *125*(1), 315–344. https://doi.org/10.32604/cmes.2020.011380

Novaković, J., Strbac, P., & Bulatović, D. (2011). Toward optimal feature selection using ranking methods and classification algorithms. *Yugoslav Journal of Operations Research*, *21*(1), 119–135. https://doi.org/10.2298/YJOR1101119N

Pang, B., & Lee, L. (2004, July). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceeding of 42nd Annual Meeting on Association for Computational Linguistics (ACL '04)* (p. 271–278). https://doi.org/10.3115/1218955.1218990

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, *2*(1–2), 1–135. https://doi.org/10.3748/wjg.v22.i45.9898

Pang, B., Lee, L., & Vaithyanathan, S. (2002, July). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 79–86).

Parlar, T., Özel, S. A., & Song, F. (2018). QER: A new feature selection method for sentiment analysis. *Human-Centric Computing and Information Sciences*, *8*(1), 1–19. https://doi.org/10.1186/s13673-018-0135-8

Rajpoot, A. K., Nand, P., & Abidi, A. I. (2021, January). A comprehensive survey on effective feature selection approaches for text sentiment classification process. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering* (pp. 971–977). IEEE.

Sainin, M. S., Alfred, R., & Ahmad, F. (2021). Ensemble Meta classifier with sampling and feature selection for data with multiclass imbalance problem. *Journal of Information and Communication Technology*, *20*(2), 103–133. https://doi.org/10.32890/jict2021.20.2.1

Shahana, P. H., & Omman, B. (2015). Evaluation of features on sentimental analysis. *Procedia Computer Science*, *46*, 1585–1592. https://doi.org/10.1016/j.procs.2015.02.088

Tang, J., Alelyani, S., & Liu, H. (2014). Feature selection for classification: A review. In C. C. Aggarwal (Ed.), *Data classification: Algorithms and applications* (p. 37). Chapman and Hall/CRC.

Wijesinghe, A. (2017). *Sentiment analysis on movie reviews.* Australian National University Technical Report–RESEARCH GATE. https://doi.org/10.13140/RG.2.2.13784.80645

Xia, R., Zong, C., & Li, S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences*, *181*(6), 1138–1152. https://doi.org/10.1016/j.ins.2010.11.023

Yousefpour, A., Ibrahim, R., & Hamed, H. N. A. (2017). Ordinal-based and frequency-based integration of feature selection methods for sentiment analysis. *Expert Systems with Applications*, *75*, 80–93. https://doi.org/10.1016/j.eswa.2017.01.009

Yousefpour, A., Ibrahim, R., Nuzly, H., & Hamed, A. (2014a). A novel feature reduction method in sentiment analysis. *International Journal of Innovative Computing*, *1*(4), 34–40.

Yousefpour, A., Ibrahim, R., Nuzly, H., & Hamed, A. (2014b, April). Feature reduction using standard deviation with different subsets selection in sentiment analysis. In *Proceedings of the 6th Asian Conference on Intelligent Information and Database Systems* (pp. 33–41). Springer, Cham. https://doi.org/10.1007/978-3-319-05458-2

Zin, H. M., Mustapha, N., Murad, M. A. A., & Sharef, N. M. (2018). Term weighting scheme effect in sentiment analysis of online movie reviews. *Advanced Science Letters*, *24*(2), 933–937. https://doi.org/10.1166/asl.2018.10661