Journal of Theoretical and Applied Information Technology
30th April 2021. Vol.99. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A METAHEURISTIC APPROACH FOR SOLVING FEATURE SELECTION IN SENTIMENT ANALYSIS PROBLEM

MOCHAMAD WAHYUDI[1], MUHAMMAD ZARLIS[2], HERMAN MAWENGKANG[3], SYAHRIL EFENDI[4]

[1]Graduate Program of Computer Science, Universitas Sumatera Utara, Indonesia

[2,4]Department of Computer Science, Universitas Sumatera Utara, Indonesia

[3]Department of Mathematics, Universitas Sumatera Utara, Indonesia

E-mail: [1]mochamad.wahyudi@students.usu.ac.id, [2]m.zarlis@usu.ac.id, [3]mawengkang@usu.ac.id, [4]syahril1@usu.ac.id

ABSTRACT

Huge business data could make data analysis becomes problematic such that the decision-making procedure would be improbable. In the topics of consumer buying behavior, an interesting technique known as sentiment analysis can support in obtaining information about the latest trends and is capable to raise market value of product through upgrading its quality. One peculiar method in solving the sentiment analysis is feature selection technique. Yet, this method includes a combinatorial behavior and the analysis of the huge data can experience difficulty in solving the combinatorial feature selection problem. In order for tackling the combinatorial problem, this paper proposes a new metaheuristic approach based on the movement of non basic variables,in such a way could force the basic non-integer variables to take integer values. The combinatorial structure of the feature selection approach for sentiment analysis can be implemented in various marketing applications.

Keywords: *Combinatorial Optimization, Sentiment Analysis, Feature Selection Approach, Buying Behavior Analysis.*

1. INTRODUCTION

During decision-making process, it is necessarily for an organization to consider the opinions from outside people to grasp critical piece of information. In this digital era the Internet and social media would become the ideal source of people's opinions. As the number of online documents growing rapidly, it would be time-consuming and challenging to dig out and analyze the wanted opinionated information. In terms of computational point of view, sentiment analysis becomes a challenging studied area for many researchers because of the bursting upsurge in the Internet users.

Sentiment analysis can be interpreted as a computational process of information in such a way could determine the customers' opinion and give decisions in the text form acquired from internet and social media. It is a decision process after identifying and categorizing the impression of customers in the text format. Therefore, in terms of data mining, it can be called as opinion mining. It could determine whether the motive of documents are either supporting or disapproving opinions. In relation with this, customers use supporting words like- 'great', 'fabulous', 'remarkable', etc., negative words would be - 'bad', 'fraud' etc. Organizations found out the customer's opinion which could be positive or negative about the product by assessing the data. Evaluation of gathered data intensely give insights into trends, while evaluation of individual cares in real companies address and swiftly answer customer concerns.

In the evaluation process, several of appropriate analyses regarding to accessible resources, datasets that are of standard and the evaluation campaigns need to be chosen. Fortunately, there are various opinion-rich resources available, such as, the online review sites and the personal blogs, appearing opportunities and the obstacles that are coming out from the extensive adaptation of information technology trends are used for labeling the opinions. The rising level in the opinion mining and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

sentiment analysis works with computation treatment of opinion, text subjectivity and sentiments. It is said that people tend to be more focused in certain aspects rather than the whole entity, theoretically opinion mining techniques could be applied to obtain particular aspects rather than the whole entity. [1] addressed in their paper about how to improve aspect-level opinion mining considering operating online customer reviews. They proposed a model of Joint Aspect/Sentiment model so as to gain aspects and the aspect dependent sentiment lexicons that could be obtained from the online customer reviews. Among the aspect dependent sentiment lexicon indicates to aspect exact opinion words consist of aspect-aware sentiment polarities concerning to specific aspects. Furthermore, the acquired aspect-dependent sentiment lexicons which are used to opinion mining duties at the aspect level comprising several aspects like aspect identification, aspect-based extractive opinion summarization and the aspect-level sentiment classification factors.

In [2] the authors suggested a model based on Sentiment Analysis and Opinion Mining from Social media. In the paper, they introduced a technique based on 3 sections in sentiment analysis such as pre-processing, feature extraction and classifications. Pre-processing process is used to increase the accuracy of the system. Pre-processing method is the most crucial aspect in Sentiment Analysis in order to obtain the accurate results. Unigrams and bigrams are used to proficient the personalities of the system in pre-processing method.

In [3] proposed a method based on Improved Feature Extraction and Classification-Sentiment Analysis. The authors focused the proposed method by comparing with the earlier existing system. To overcome all the difficulties in the existing system the authors suggested a machine learning classification to achieve the feature extraction model will be more efficient and proficient one. Then, the following categories are based on feature-based sentiment analysis such as, feature extraction, sentiment classification and sentiment evaluation.

Feature selection techniques are applied to rank characteristics so that non-informative characteristics can be eliminated to develop the classification performance [4]. Some researchers have investigated the effects of feature selection for sentiment analysis [5]–[8]. The objective of feature selection is to choose the most related and distinguishing features through the removal of features that are unrelated for classification

2. SENTIMENT ANALYSIS

Sentiment analysis or opinion mining refers to a wide range of fields of natural language processing, linguistic computation and text mining, with the aim of analyzing the opinions, feelings, assessments, attitudes, judgments and emotions of a person, whether the speaker or the writer is related to a subject, a service product, an organization, an individual or certain activities [9]. The purpose of the sentiment analysis is to determine the behavior or opinion of a writer by paying attention to a particular subject. Behavior may indicate reasons, opinions or judgments, trend conditions [10]. Sentiment analysis may also express emotional feelings of sadness, joy or anger. In general, it is found that the steps in the text classification of sentiment analysis according to [11] are as follows:

1. The data set for the domain should be defined, which involves collecting datasets that cover a domain
2. Initial pre-processing stage or tokenization process
3. Transformation
   The process of representing numbers calculated on the basis of text data. Binary representation is generally used and counts only the presence or absence of a word in a document. The number of times a word appears in a document is also used as a weighting scheme for text data. The commonly used processes are TF-IDF, Binary Transformation and Frequency Transformation.
4. Feature Selection
   Feature selection can make classifiers more efficient or effective by reducing the amount of data to be analyzed by identifying the relevant features that will be further processed. Feature selection methods that are usually used include: expert knowledge, minimum frequency, information gain, chi square, and so on.
5. Classification
   Classification processes generally use classifiers such as, Naïve Bayes, Support Vector Machine (SVM) and others.
6. Interpretation or evaluation
   The evaluation stage usually calculates accuracy, recall, and accuracy.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. FEATURE SELECTION TECHNIQUE

In classification, removal of the relevant feature decreases classification performance. The presence of irrelevant features also affects classification performance [12]. Redundancy among features found by calculating the correlation between features [13] and removal of redundant features improves classification performance. In feature generation feature numbering in the thousands is made and usually a large number of these features do not provide any information because of irrelevance or special class redundancy and must be removed to improve classification performance [12], [14]. A feature is said to be extraordinary if it is very good at adjusting the predictive ability of the classifier in assigning new samples to the class in sentiment analysis [15].

Machine learning model performance depends not only on a sophisticated approach to feature extraction and weights assignment, but also on efficient feature selection techniques. The aim of the feature selection technique is to select the most relevant and discriminatory features by removing the noisy and irrelevant features for classification. If the higher length of feature vectors and noisy and irrelevant features [16] are present, the performance of the machine learning process is reduced.

In order to reduce the dimensions of feature vector machine learning techniques, such as information gain [17], mutual information [17], etc., or transformation techniques, such as singular value decomposition, LDA, etc., are used. The feature selection method selects important features based on the value of the term and selects the best features and reduces the remaining irrelevant features. The transformation method of a feature is based on converting a high-dimensional vector feature into a low-dimensional space where a reduced vector feature like the previous vector feature is generated from the contribution of each feature. In the research [18], dimension reduction was carried out using latent semantic analysis techniques and the vector reduction feature of the support vector machine was used to improve the performance of sentiment analysis. A popular method for reducing the length of a feature vector is the selection of an appropriate feature for transformation due to its simplicity and computational efficiency.

Several features of selection techniques such as: mutual information, information gain, document frequency, chi-square, etc. in [14], [15], [19], [20] have attempted sentiment analysis.

The simplest feature selection technique is the frequency of the document. Document Frequency is a sentiment analysis method that is usually used in [21] research using terms that most often appear in the text used to create feature vectors. [15] tested the performance of four feature selection techniques (mutual information, information gain, document frequency and chi-square) in the Chinese document sentiment analysis. The research uses five machine learning algorithms, such as: centroid classifier, K-Nearest Neighbor, Naïve Bayes, winnow classifier, and vector support. Experimental results show that information gain performance in the selection of features is better than other approaches and that the performance of the support vector machine algorithm is the best.

[22] conducted information gain and genetic algorithm research using a film review dataset and suggested a hybrid method called an entropy weighted genetic algorithm to improve the accuracy of sentiment analysis. In this study [6], a new method for the selection of features called document frequency differences was developed and a comparison of the suggested methods with other feature selection techniques for sentiment analysis was developed. In a study [23] to conduct a sentiment analysis, the log-likelihood method was used to select key features.

In sentiment analysis it is necessary to process the selecting features that are relevant to the response variable under consideration. This process is called feature selection. As there are many simple and easy of intelligent techniques are available then it is not surprising that these techniques have been widely used in the process of features selection [24]–[26].

Feature selection, particularly for sentiment analysis, is a challenging task as one can face with complex interaction among features. A separate relevant feature may turn out to be irrelevant when the selecting process is carried out simultaneously with other features. Therefore, to eliminate the negative impact of the irrelevant and redundant features, various feature selection methods have been proposed. The main target of feature selection is to choose relevant features from a large feature set [4]. From the structure of the problem, assuming that the problem want to be solved optimally, feature selection turns out as a difficult combinatorial optimization problem [27]. From practical point of view, there are two intractable conditions could happen. Firstly, it is very unlikely to find out the size of the feature subset, then the impact is the dimensionality of the decision space is non-reducible. Secondly, the fact that in opinions the features could have complementary or contradictory interactions with each other, then

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

regarding to solving procedure in optimization problem, the decision space is non-separable. Thus, given a feature set with m-dimension, the number of all possible feature subsets would be as large as 2^m, then to select relevant features using traditional exhaustive search approaches would be practically impossible. To tackle these difficulties in relating to solving complex combinatorial optimization problems, there are numbers of metaheuristic approach have been proposed, viz., genetic algorithms (GAs) [28], differential evolution (DE) [29], and particle swarm optimization (PSO) [30], among many others [31]. Among these metaheuristic approach, PSO turns out to be well known due to its algorithmic simplicity and computationally less expensive. There are many authors who have proposed to apply PSO to solve feature selection problems [32], [33]. However, the original PSO has many limitations for feature selection [34]. Firstly, PSO was initially proposed for continuous optimization problems, while feature selection is a combinatorial optimization problem. Secondly, although PSO shows promising performance on low-dimensional problems, it suffers for solving big dimension problems [35]. In their paper [36] have proposed a PSO variant, known as competitive swarm optimizer (CSO), for large-scale optimization [37]. PSO with golden ratio principle was proposed for tackling feature selection [38].

4. COMBINATORIAL OPTIMIZATION

In the development of discrete mathematics and computational mathematics, the term combinatorial optimization is increasingly being developed and can be applied in a variety of fields, particularly in the search and arrangement of objects, without having to list all possible arrangements. Combinatorial optimization is also a method used to find all possible actual values of the objective function. The search process can be done by registering one by one of the possible values by developing a search algorithm. The best one is chosen in the search process using one of the search methods. In other words, combinatorial optimization is based on the maximum or minimum value, depending on the problem presented, and combinatorial optimization is also used to solve problems that are quite complex and of sufficient scope.

Combinatorial problems are problems that have a finite set of feasible solutions. Although, in principle, the solution to this problem can be achieved by a complete listing, complex problems require time which cannot be accepted in practice [39]. According to [40], combinatorial analysis is a mathematical study of the arrangement, grouping, ordering, or selection of discrete objects, usually finite in number, many combinatorial optimization problems have arisen from research in computer design, computational theory, and computer applications in numerical problems requiring new methods, new approaches, and new mathematical approaches.

5. METAHEURISTIC ALGORITHM

Search based software engineering is one of interesting subjects in the last five years [41]. Search-based software engineering has been successfully used to solve various problems of software engineering in design, testing, software engineering, requirements engineering and refactoring. Search-based software engineering involves applying search algorithms to solve software engineering problems. In search-based software engineering, software engineering problems are reformulated as search-based optimization problems or search-based problems [42], [43].

Software engineering problems can be formulated as problems of optimization, therefore metaheuristic algorithms can be used to solve such problems [44], [45]. Examples of successful use of metaheuristic algorithms for software engineering problems include software design, project planning and cost estimation, requirements engineering, quality assessment, automated maintenance, compiler optimization, software aesthetic services,oriented software engineering, reverse engineering, software measurement, and software testing [44], [46], [47].

A new type of approximation algorithm has emerged in the last 19 years and seeks to combine basic heuristic methods in search or optimization methods to explore the search space, a method commonly referred to as the metaheuristic method [48]. The metaheuristic method is described as an excellent alternative search strategy, above the search space in the hope of finding the best results first introduced in 1986. The algorithms proposed by the researchers include the ant colony optimization method [49]. Until now, there is no generally accepted definition and standard reference for the term metaheuristic, but metaheuristic is usually used as a high-level strategy that addresses the underlying problem. Metaheuristic is formally defined as an iterative generating process that guides subordinate heuristics by intelligently combining different

Journal of Theoretical and Applied Information Technology
30th April 2021. Vol.99. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

concepts to explain and exploit the search space, learning strategies are used to organize information in such a way that it can be found efficiently in the vicinity of the optimal solution [50].

Metaheuristic algorithms have been developed, such as: TS [51], SA [52], GA [53], ACA [54], PSO [55], differential evolution [56], HS [57], FPA [58], sine cosine algorithm [59], BA [60], CS [61] and firefly algorithm [62].

Glover [51] first introduced a metaheuristic algorithm in the TS algorithm, which characterized it as an attempt to escape from a local optima. The metaheuristic algorithm is part of a stochastic algorithm that efficiently explores the search space through "trial and error" Unlike conventional algorithms, most metaheuristic algorithms are

considered population-based algorithms where finding a solution begins from a variety of positions in the solution space. Every member of the population is a candidate for the best solution possible. The metaheuristic algorithm provides guidance on the search for space movement to explore the overall search space efficiently. The search guide is in the form of a fitness function (only a problem) where each solution has a score determined on the basis of its quality. Here, the fitness function can maximize or minimize certain parameters on the basis of the problem at hand

The metaheuristic algorithm has two key components, namely: intensification (local search) and diversification (global search). Intensification is exploring promising neighboring areas in the hope of finding better solutions. Diversification, on the other hand, ensures that all areas of the search space have been visited, allowing the algorithm to jump out of the local optimum. Balancing the interactions between the two components may significantly affect the efficiency of the metaheuristic algorithm [5].

The performance of the metaheuristic algorithm is highly dependent on a good balance between the two components. Too much intensification can result in a rapid loss of diversity in the population, which increases the likelihood of trapping the algorithm at an optimum local level. Aggressive diversification can lead to inefficient search and slow overall search performance [63].

In search-based software engineering, different metaheuristic algorithms have been used. These algorithms can be categorized into four categories: local search, swarm intelligence, evolutionary, and hybrid algorithms, as shown in Figure 1 [64]. As the name suggests, local search algorithms are often used with bias. In particular, local search algorithms usually use the knowledge of their neighbors to come up with newer solutions. The swarm intelligence algorithm uses the collective behavior of the population as a means of exploring and exploiting its search space. Evolutionary algorithms often use biological evolution, such as reproduction, mutation, recombination and selection, to come up with newer solutions. The hybrid algorithm adopts a combination of other algorithms so that it has different ways to find newer solutions.

Metaheuristic is usually a high-level strategy that guides a specific underlying heuristic problem to improve its performance, the main goal of which is iterative improvement and algorithm development.

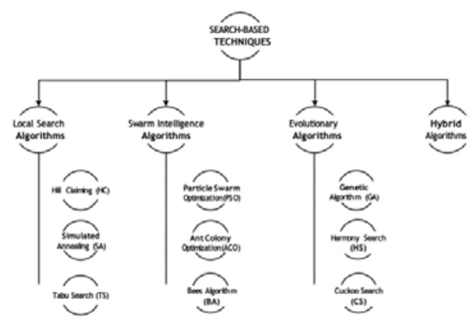

Figure 1. Metaheuristic Algorithm

Many metaheuristic approaches rely on probabilistic decisions taken during the search, but the main difference from metaheuristic searches is that they are intelligently carried out.

Thus, it can be said that metaheuristic is an iterative master process that guides and modifies the operations of subordinate heuristics in order to produce high-quality solutions efficiently. It can manipulate a single complete (or incomplete) solution or a collection of solutions at each different iteration with the heuristic method is a technique designed to solve a problem regardless of whether the solution of the problem can be proven correct or not, but usually produces a good solution or solves problems more easily, quickly and simply. Heuristic techniques do not have a definite optimum solution search algorithm, but have rules that can explore the most promising search spaces, namely the space where an optimum or near-optimal solution is available [65].

Various types of heuristic methods have developed rapidly in academia. From these methods, there are

Journal of Theoretical and Applied Information Technology
30th April 2021. Vol.99. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

several methods which are born inspired by the behavior of living things, one of which is the behavior of a group of ants. The behavior of a group of ants out of their nest at the food source by leaving a pheromone substance creates an algorithm called ant colony optimization

6. COMBINATORIAL MODEL FOR FEATURE SELECTION

In our proposed method for solving sentiment analysis problem, mathematically feature selection is formulated as the following combinatorial minimization problem:

$$\underset{x}{Minimize}\ f(x)$$
$$subject\ to \quad x \in X$$

where $X \in \{0,1\}^N$ define the feasible solution set with N dimension

The selected feature sets are denoted by X which can be encoded by a number of N binary bits. That is to say that N is the total number of features in the original feature set. For each bit in x , '1' and '0' denotes that the corresponding feature is or is not selected, respectively. In this way, feature selection becomes a combinatorial optimization problem where the objective is to find the best feature subset x^* such that to minimize the error rate of the classification models (or to minimize the number of feature)

7. BASIC IDEA

The basic ideas in this study are as follows:

1. Determine the decision solution space (feasible integer solution region).
2. Reduce the region by eliminating areas that do not comply with feasible integer requirements.
3. Identify space that can provide an optimal feasible integer solution.
4. From the point obtained in Step 2, obtain a gradient direction for the area obtained in Step 3.
5. Calculate the distance of the movement along the direction vector obtained in Step 4 in such a way that the point is still feasible.
6. Check the point obtained in step 5 whether it was a sub optimal feasible solution.
7. If Yes, proceed to Step 8.

Otherwise, go back to Step 4

8. Continue the movement of the point obtained in step 5 by reducing the area so that the optimal integer feasible solution is obtained.
9. Stop

8. DESCRIPTION OF THE PROPOSED APPROACH

Define the feasible region for the solution of the original optimization problem. Given that ***g(X)*** is the constraint function of the optimization problem with $g := R^n \rightarrow R$ then the set $\{x : g(x) = b\}$, with b is a constant, is called the feasible region or the optimization problem solution space. For combinatorial optimization, function is defined as a mapping g:{0,1}^n→{0,1}, Noted that set {x:g(x)=b} is a set of feasible points. Let set S={x:g(x)=b}, if there is an x∈S then x is called a feasible point.

8.1. Determination of a feasible point

Suppose the general form of a combinatorial optimization problem can be written as follows.

Maximize *Z=f(x)* *(P)*

Subject to *g_i (x)=b_i,i=1,2,…,m*

x∈{0,1}

If and is linear then problem (P) can be stated in the form of

Maximize Z=C^T X
Subject to Ax=b (P0)
x∈{0,1}

Binary terms are relaxed such that (PO) can be written as

Maximize Z=C^T X
Subject to Ax=b (P1)
X≥0

The constraint matrix *A m x n* (*m* number of rows, *n* number of columns) can be partitioned into a basic matrix (B) with size *m x m*, and a nonbasic matrix *(N)* with size *(m x (n-m)).*
Then, we can write

A=(BN) (1)

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Analog with vectors of variable x can be partitioned accordingly with vector XB as the basic variable and XN as the non-basic variable.
Now the expression
Ax=b
becomes
(BN)((X_B@X_N))=b (2)
BX_B+NX_N=b (3)
or
BX_B=b-NX_N (3)

Because matrix B is a basic matrix, it has an element (B-1).
Multiply from the left of the Equation (3) with B-1

B(-1) BX_B=B(-1) b-B(-1) NX_N (4)
IX_B-B(-1) b-B(-1) NX_N (5)

We get,
X_B=B^(-1) b-B^(-1) NX_N (6)

Value of *Xn* is 0, from *x>0* .
Therefore we obtain a feasible solution point for (P1) as
X_B=β
With
$\beta=B^{-1}$ b

8.2 Optimization Test

After the feasible point that has been obtained, (Eq. 6) needs to be tested whether this point is optimal for the problem (P1) or not.
Consider the objective function of (P1)

$Z=C^T X$
Vectors C and x are partitioned according to the basic matrix (B) and non-basic matrix (N)
Then Z=(C_B C_N) (X_B@X_N)) (7)
SCENARIO TREE

$$Z = (C_B C_N)\begin{pmatrix} X_B \\ X_N \end{pmatrix}$$

Then
$Z_{\_j}-C_{\_j}=C^B B^{-1} N-C_{\_N}$ (12)

If $Z_j-C_j<0$ (negative), the *XN* vector is raised from the boundary 0, the value of *Z* is increased for the problem (P1), which means that the value of the objective function can be increased. In other words, the XB point obtained has not yet resulted as a maximum solution. However, if $Z_j-C_j \geq 0$, the vector XN is raised from its boundary 0, the value of Z will decrease (shrink) or remain.

This means that the XB point obtained is already the maximum point, so that the maximum conditions for the problem (LP) have been obtained, namely:
$Z_j-C_j \geq 0 \ \forall j$ (non-basic) (13)

8.3 Feasible point for Combinatorial Optimization

Note that the feasible value point expressed in Equation (6) is for the problem (P1).
Now, consider the problem (PO).

Constraints x∈{0,1} can be expressed in the form of

$$x \geq 0$$
$$x \leq 1$$

As a result, the problem (PO) can be written as
Maximize

$$Z=C^T X$$

Constraint

$$AX=b$$
$$x \geq 0$$
$$x \leq 1$$

Analog with the problem (P1), a feasible point value also takes the form of Eq. (6), namely:

$$X_B=B^{-1} b-B^{-1} N_{XN}$$
$$X_B=\beta-\alpha X_N \quad (14)$$

With *β=B-1 b and α=B-1 N*

In the problem (P1) XN is 0 because of the non-negativity condition, in other words the value of the variable XN is 0 at the boundary point.
Then for combinatorial problems (PO) which has

$$x \geq 0$$
$$x \leq 1$$

Then the value of the non basic variable *XN* is 0 or 1, it means that the value of xn is already binary.
If all components of the vector β have a value of 0 or 1, this means that a feasible integer solution to the problem (PO) has been found. If the component of the vector β still has no 0 or 1 it means that the feasible solution to the problem (PO) has not been achieved.

The procedure for obtaining binary values is as follows:

1. Separate the set of basic variables I1 from the set I2, these variables are the bases within the boundary of 0 or 1 and the set I1 for the sets I2, I3.

2. Search for the objective function by maintaining the basic variable I1 and only discrete changes to the value of the variables in the I2 set.

When the Step 2 is accomplish, check the value of Z_j-C_j from Eq. (12) for the variable in set I1. If something can be moved from the boundary, add to the I2 set, repeat from the Step 2, if it nothing can be moved, then stop. The result of this procedure is that all components of the β vector have a binary value so that a basic feasible integer solution to the problem (PO) is achieved.

9. CONCLUSIONS

This paper presents feature selection technique modeled as a combinatorial optimization problem for sentiment analysis. We propose a metaheuristic approach for solving the problem. The basic idea of the approach is to explore the resulted continuous solution space which contains feasible integer solution points to the combinatorial optimization problem.

REFRENCES:

[1] X. Xu, X. Cheng, S. Tan, Y. Liu, and H. Shen, "Aspect-level opinion mining of online customer reviews," *China Commun.*, vol. 10, no. 3, pp. 25–41, 2013.

[2] S. Mathapati and S. H. Manjula, "Sentiment analysis and opinion mining from social media: A review," *Glob. J. Comput. Sci. Technol.*, 2017.

[3] M. Trupthi, S. Pabboju, and G. Narasimha, "Improved feature extraction and classification—Sentiment analysis," in *2016 International Conference on Advances in Human Machine Interaction (HMI)*, 2016, pp. 1–6.

[4] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[5] D.-H. Yang and G. Yu, "A method of feature selection and sentiment similarity for Chinese micro-blogs," *J. Inf. Sci.*, vol. 39, no. 4, pp. 429–441, 2013.

[6] C. Nicholls and F. Song, "Comparison of feature selection methods for sentiment analysis," in *Canadian Conference on Artificial Intelligence*, 2010, pp. 286–289.

[7] F. Akba, A. Uçan, E. A. Sezer, and H. Sever, "Assessment of feature selection metrics for sentiment analyses: Turkish movie reviews," in *8th European Conference on Data Mining*, 2014, vol. 191, no. 2002, pp. 180–184.

[8] Y. Liu, J.-W. Bi, and Z.-P. Fan, "Multi-class sentiment classification: The experimental comparisons of feature selection and machine learning algorithms," *Expert Syst. Appl.*, vol. 80, pp. 323–339, 2017.

[9] B. Liu, "Sentiment analysis and opinion mining," *Synth. Lect. Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012.

[10] A. S. H. Basari, B. Hussin, I. G. P. Ananta, and J. Zeniarja, "Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization," *Procedia Eng.*, vol. 53, pp. 453–462, 2013.

[11] R. Moraes, J. F. Valiati, and W. P. G. Neto, "Document-level sentiment classification: An empirical comparison between SVM and ANN," *Expert Syst. Appl.*, vol. 40, no. 2, pp. 621–633, 2013.

[12] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *J. Mach. Learn. Res.*, vol. 5, no. Oct, pp. 1205–1224, 2004.

[13] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, 2005.

[14] B. Agarwal and N. Mittal, "Optimal feature selection for sentiment analysis," in *International Conference on Intelligent Text Processing and Computational Linguistics*, 2013, pp. 13–24.

[15] S. Tan and J. Zhang, "An empirical study of sentiment analysis for chinese documents," *Expert Syst. Appl.*, vol. 34, no. 4, pp. 2622–2629, 2008.

[16] Y. Aphinyanaphongs *et al.*, "A comprehensive empirical comparison of modern supervised classification and feature selection methods for text categorization," *J. Assoc. Inf. Sci. Technol.*, vol. 65, no. 10, pp. 1964–1987, 2014.

[17] C. D. Manning, H. Schütze, and P. Raghavan, *Introduction to information retrieval*. Cambridge university press, 2008.

[18] L. Wang and Y. Wan, "Sentiment classification of documents based on latent semantic analysis," in *International*

Conference on Computer Education, Simulation and Modeling, 2011, pp. 356–361.

[19] B. Agarwal and N. Mittal, “Text classification using machine learning methods-a survey,” in *Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012), December 28-30, 2012*, 2014, pp. 701–709.

[20] B. Pang and L. Lee, “Opinion Mining and Sentiment Analysis,” *Found. Trends® Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008.

[21] B. Pang, L. Lee, and S. Vaithyanathan, “Thumbs up? Sentiment classification using machine learning techniques,” *arXiv Prepr. cs/0205070*, 2002.

[22] A. Abbasi, H. Chen, and A. Salem, “Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums,” *ACM Trans. Inf. Syst.*, vol. 26, no. 3, pp. 1–34, 2008.

[23] M. Gamon, “Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis,” in *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, 2004, pp. 841–847.

[24] O. S. Qasim and Z. Y. Algamal, “Feature selection using particle swarm optimization-based logistic regression model,” *Chemom. Intell. Lab. Syst.*, vol. 182, pp. 41–46, 2018.

[25] X. Wang, J. Yang, X. Teng, W. Xia, and R. Jensen, “Feature selection based on rough sets and particle swarm optimization,” *Pattern Recognit. Lett.*, vol. 28, no. 4, pp. 459–471, 2007.

[26] D. Rodrigues *et al.*, “A wrapper approach for feature selection based on bat algorithm and optimum-path forest,” *Expert Syst. Appl.*, vol. 41, no. 5, pp. 2250–2258, 2014.

[27] I. A. Gheyas and L. S. Smith, “Feature subset selection in large dimensionality domains,” *Pattern Recognit.*, vol. 43, no. 1, pp. 5–13, 2010.

[28] K.-H. Han and J.-H. Kim, “Quantum-inspired evolutionary algorithm for a class of combinatorial optimization,” *IEEE Trans. Evol. Comput.*, vol. 6, no. 6, pp. 580–593, 2002.

[29] K. Price, R. M. Storn, and J. A. Lampinen, *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media, 2006.

[30] A. Banks, J. Vincent, and C. Anyakoha, “A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications,” *Nat. Comput.*, vol. 7, no. 1, pp. 109–124, 2008.

[31] M. Iqbal, M. Zarlis, T. Tulus, and H. Mawengkang, “Model Pendekatan Metaheuristik Dalam Penyelesaian optimisasi Kombinatorial,” in *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)*, 2020, vol. 1, no. 1, pp. 92–97.

[32] A. Unler and A. Murat, “A discrete particle swarm optimization method for feature selection in binary classification problems,” *Eur. J. Oper. Res.*, vol. 206, no. 3, pp. 528–539, 2010.

[33] L.-Y. Chuang, S.-W. Tsai, and C.-H. Yang, “Improved binary particle swarm optimization using catfish effect for feature selection,” *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12699–12707, 2011.

[34] B. Xue, M. Zhang, and W. N. Browne, “Particle swarm optimization for feature selection in classification: A multi-objective approach,” *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1656–1671, 2012.

[35] B. Xue, M. Zhang, and W. N. Browne, “Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms,” *Appl. Soft Comput.*, vol. 18, pp. 261–276, 2014.

[36] Y. Zhai, Y.-S. Ong, and I. W. Tsang, “The emerging" big dimensionality",” *IEEE Comput. Intell. Mag.*, vol. 9, no. 3, pp. 14–26, 2014.

[37] R. Cheng and Y. Jin, “A competitive swarm optimizer for large scale optimization,” *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 191–204, 2014.

[38] K. Saravanapriya and Bagyamani, “A Rough Set Pooled Fitness Function Based Particle Swarm Optimization Algorithm using Golden Ratio Principle for Feature Selection,” *Int. J. Eng. Adv. Technol.*, vol. 9, no. 1, pp. 3785–3790, Oct. 2019.

[39] J. Lee, *A first course in combinatorial optimization*, no. 36. Cambridge University Press, 2004.

[40] E. Lawler, “Combinatorial optimization. Networks and matroids. Repr.,” p. x + 374, 2001.

[41] M. Harman, Y. Jia, J. Krinke, W. B.

Langdon, J. Petke, and Y. Zhang, “Search based software engineering for software product line engineering: a survey and directions for future work,” in *Proceedings of the 18th International Software Product Line Conference-Volume 1*, 2014, pp. 5–18.

[42] M. Harman, Y. Jia, and Y. Zhang, “Achievements, open problems and challenges for search based software testing,” in *2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST)*, 2015, pp. 1–12.

[43] M. Harman and P. McMinn, “A theoretical and empirical study of search-based testing: Local, global, and hybrid search,” *IEEE Trans. Softw. Eng.*, vol. 36, no. 2, pp. 226–247, 2009.

[44] J. Clarke *et al.*, “Reformulating software engineering as a search problem,” *IEE Proceedings-software*, vol. 150, no. 3, pp. 161–175, 2003.

[45] M. Harman and B. F. Jones, “Search-based software engineering,” *Inf. Softw. Technol.*, vol. 43, no. 14, pp. 833–839, 2001.

[46] M. Harman, “The current state and future of search based software engineering,” in *Future of Software Engineering (FOSE’07)*, 2007, pp. 342–357.

[47] M. Harman, S. A. Mansouri, and Y. Zhang, “Search-based software engineering: Trends, techniques and applications,” *ACM Comput. Surv.*, vol. 45, no. 1, pp. 1–61, 2012.

[48] A. Alazzam, E. Yuzgec, and H. W. Lewis III, “A new optimization algorithm for non-convex problems,” in *IIE Annual Conference. Proceedings*, 2013, p. 2784.

[49] C. Blum and A. Roli, “Metaheuristics in combinatorial optimization: Overview and conceptual comparison,” *ACM Comput. Surv.*, vol. 35, no. 3, pp. 268–308, 2003.

[50] I. H. Osman and G. Laporte, “Metaheuristics: A bibliography.” Springer, 1996.

[51] F. Glover, “Tabu search—part I,” *ORSA J. Comput.*, vol. 1, no. 3, pp. 190–206, 1989.

[52] S. Kirkpatrick, “Optimization by simulated annealing: Quantitative studies,” *J. Stat. Phys.*, vol. 34, no. 5, pp. 975–986, 1984.

[53] J. H. Holland, “Genetic algorithms,” *Sci. Am.*, vol. 267, no. 1, pp. 66–73, 1992.

[54] R. N. Khushaba *et al.*, “Ant Colony Optimization and Swarm Intelligence: 6th International Conference, ANTS 2008, Brussels, Belgium, September 22-24, 2008. Proceedings,” 2008.

[55] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95-international conference on neural networks*, 1995, vol. 4, pp. 1942–1948.

[56] V. Feoktistov, *Differential evolution*. Springer, 2006.

[57] K. S. Lee and Z. W. Geem, “A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice,” *Comput. Methods Appl. Mech. Eng.*, vol. 194, no. 36–38, pp. 3902–3933, 2005.

[58] X.-S. Yang, “Flower pollination algorithm for global optimization,” in *International conference on unconventional computing and natural computation*, 2012, pp. 240–249.

[59] S. Mirjalili, “SCA: a sine cosine algorithm for solving optimization problems,” *Knowledge-based Syst.*, vol. 96, pp. 120–133, 2016.

[60] D. T. Pham, A. Ghanbarzadeh, E. Koç, S. Otri, S. Rahim, and M. Zaidi, “The bees algorithm—a novel tool for complex optimisation problems,” in *Intelligent production machines and systems*, Elsevier, 2006, pp. 454–459.

[61] X.-S. Yang and S. Deb, “Cuckoo search via Lévy flights,” in *2009 World congress on nature & biologically inspired computing (NaBIC)*, 2009, pp. 210–214.

[62] X.-S. Yang, “Firefly algorithm, Levy flights and global optimization,” in *Research and development in intelligent systems XXVI*, Springer, 2010, pp. 209–218.

[63] X.-S. Yang, S. Deb, and S. Fong, “Metaheuristic algorithms: optimal balance of intensification and diversification,” *Appl. Math. Inf. Sci.*, vol. 8, no. 3, p. 977, 2014.

[64] R. Malhotra and M. Khanna, “Common threats to software quality predictive modeling studies using search-based techniques,” in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2016, pp. 554–560.

[65] R. J. Thierauf and R. A. Grosse, “Decision making through operations research,” 1970.