Feature selection based on chi-square and ant colony optimization for multi-label classification

Joan Angelina Widians[1,3], Retantyo Wardoyo[2], Sri Hartati[2]
[1]Doctoral Program of Computer Science, Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Gadjah Mada University, Yogyakarta, Indonesia
[2]Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Gadjah Mada University, Yogyakarta, Indonesia
[3]Department of Informatics, Faculty of Engineering, Mulawarman University, Samarinda, Indonesia

Article Info

Article history:

Received Oct 1, 2023
Revised Feb 26, 2024
Accepted Feb 27, 2024

Keywords:

Ant colony optimization
Chi-square
Feature selection
Machine learning
Multi-label classification

ABSTRACT

Text classification is widely used in organizations with large databases and digital documents. In text classification, there are many features, most of which are redundant. High-dimensional features impact multi-label classification performance. Feature selection is a data processing technique that can overcome this problem. Feature selection techniques have two major approaches: filter and wrapper. This paper proposes a hybrid filter-wrapper technique combining two algorithms: Chi-square (CS) and ant colony optimization (ACO). In the first stage, CS is used to reduce the number of irrelevant features. The ACO method is in the second stage. The ACO is applied to select the efficient features and improve classifier performance. The experiment results show that CS-ACO, CS-grey wolf optimizer (GWO), CS, and without feature selection (FS) have a micro F1-score based multinomial naïve Bayes classifier including 80%, 79.75%, 79.64% and 77.78%. The result indicates that the CS-ACO algorithm is suitable for solving multi-label classification problems.

Corresponding Author:

Retantyo Wardoyo
Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Gadjah Mada University
North Sekip, Bulak Sumur, Yogyakarta, 55281, Indonesia
Email: rw@ugm.ac.id

1. INTRODUCTION

The abundance of information on the internet poses a challenge for data analysis. Text document classification is widely used to handle volumes of data in cases where each instance has multiple labels assigned to it (known as multi-label classification (MLC)). In many cases and labelling situations, it becomes crucial to identify the relevant and informative features that accurately predict the various categories for each instance [1]. However, the vast number of features in this field makes text classification challenging. It is essential to select the features that are genuinely relevant to the classification process, and irrelevant ones could significantly impact the accuracy of the classifiers [2]. Therefore, the feature selection (FS) is necessary to obtain characteristics that describe document content, reduce feature space complexity, and improve performance.

FS selects features as part of the overall set of features that can be considered for exploring state space. The entire space is investigated using a comprehensive search strategy or heuristic. A complete search strategy can only be applied when we have few features [3]. In heuristic search, the features that still need to be selected will be considered at each step of the evaluation process. Random subsets are generated using a haphazard search approach. FS is divided into two approaches, namely Wrapper and filter [4], [5].

In the wrapper method, inserting and removing parts is carried out based on the accuracy of the classifier to obtain selected feature candidates. Typically, superior classification results are achieved with the wrapper method compared to the filter method. The filter approach uses statistical measurements to evaluate the relevance and importance of features independent of any particular machine learning (ML) algorithm. ML is a subfield of computer science focused on theory, performance, and learning algorithms [6]. However, embedded-based approaches use features with less computational cost. In addition, this approach incorporates FS methods into the classifier training process (learning process) without using search algorithms such as metaheuristic algorithms. From the above description, different FS algorithms have advantages and disadvantages.

One of the most popular filter methods is the Chi-square (CS), which measures the statistical significance of the relationship between features and class labels in a contingency table. Regardless, CS needs to address this issue of feature redundancy, where highly correlated features are selected simultaneously, which leads to overfitting and reduced generalization performance [7]. The wrapper method directly incorporates machine learning algorithms into the FS [8]. The wrapper approach is popular in FS, where ML algorithms evaluate the interaction between features and the efficiency of different feature subsets. The wrapper uses a search strategy to select a subset of features that optimizes the performance of a given ML algorithm [9], [10].

Evolutionary algorithms have been utilized as an FS method by many researchers, such as swarm intelligence (SI) algorithms [11], [12]. SI is an algorithm for optimization problems that imitate animal behavior and natural life. Examples of SI are ant colony optimization (ACO) [13], particle swarm optimization (PSO) [14], cuckoo optimization algorithm (COA) [15], grey wolf optimizer (GWO) [9], [16], and more. Marco Dorigo presented the ACO algorithm based on the behavior of ants in the early 1990s [17]. ACO was inspired by real-life observations of ants searching for the shortest routes to food. It could solve the significant high-dimensional feature space problem in text classification [18], [19].

In contrast, Mirjalili *et al.* [20] proposed the GWO algorithm in 2014. The GWO algorithm was inspired by the behaviour of the grey wolf (*Canis lupus*), especially its unique hunting techniques and hierarchy [20], [21]. The grey wolf is regarded as an apex predator, which places it at the summit of the food chain system [22]. On average, grey wolves prefer to reside in packs of five to twelve individuals. The wolf's particular hierarchy consists of four groups: alpha, beta, delta, and omega [23]–[25].

This research develops a hybrid filter-wrapper FS approach. We propose a mixed technique that combines CS and ACO for FS. The proposed method adopts CS as a filter-based approach and ACO as a wrapper approach. Our study consisted of 2 stages of combined FS. In the first stage, CS is used as a filter to reduce the number of features. In the second phase, the ACO method is applied as a wrapper approach to identify feature subsets with the highest classification performance to build a classification model. The ACO algorithm is used to overcome performance problems that CS methods cannot. We apply the classifier chain (CC) to the MLC problem. We use ML algorithms such as multinomial naïve Bayes (MNB), complement naïve Bayes (CNB), and linear support vector classifier (LSVC) to evaluate the performance of hybrid FS techniques. This paper is divided into the following sections. Related work in section 2. Section 3 describes the research method. Section 4 analyzes the results of the CS-ACO experiment. A summary of the article is provided in section 5.

2. LITERATURE REVIEW

The SI approach optimizes a wrapper model's feature subset selection process. Evaluating the quality of a particular subset of features requires continuously applying computationally expensive ML techniques. This SI algorithmic approach for FS seeks to identify which subset of all available features is paired with a preset. The Wrapper FS technique reduces the search space open to find elements. This technique searches for a subset using the best algorithm before the classifier process. In addition, it also offers learning strategies for feature evaluation. Thus, attributes are selected based on their influence on increasing accuracy. The wrapper approach generally outperforms the filter approach regarding classification accuracy, especially when only a few features are available. The disadvantage of the wrapper approach is its high computational complexity. These approaches can be combined to circumvent their limitations and exploit their advantages in a hybrid approach, which many researchers have utilized [6].

The ACO algorithm is based on the behavior of ants looking for the shortest path to get food and their adaptation to natural changes. Initially, the ACO algorithm was aimed at the traveling salesman problem (TSP). Since then, many studies have used ACO on various complex topics, such as quadratic assignment problems, routing in telecommunications networks, graph coloring problems, scheduling, and more. Additionally, ACO has been successfully used in many types of research to select the final feature set [26].

In contrast, Mirjalili *et al.* [20] proposed the GWO algorithm in 2014. The algorithm was inspired by the grey wolf (*Canis lupus*) behavior, especially its hunting techniques and social hierarchy. Grey wolf is regarded as an apex predator, which places it at the summit of the food chain system. On average, grey wolves prefer to reside in packs of five to twelve individuals. The wolf social hierarchy consists of four groups: alpha wolf, beta wolf, delta wolf, and omega wolf. Alpha wolf (α) is a decision-maker and commander. Beta (β) assists the alpha in collective leadership. Delta (δ) follows the instructions of alpha and beta. The remaining wolves are omega (ω) and monitor the other wolves' movements. The grey wolf's intelligence, leadership, and hunting propensities in the wild have been the primary sources of inspiration for GWO [27]. Another work outlined a comparative study between light stemming and hard stemming on the one hand and the proposed CS and FS methods on the other hand. Accordingly, they analyzed the impact of controlling and FS on Arabic text classification regarding recall actions using decision trees. The results show that combining the proposed FS and light stemming techniques improves Arabic text classification performance [7].

3. METHOD

This paper presents a two-stage FS method for text classification. In the first stage, the CS method is applied to reduce irrelevant features. The second stage uses the ACO algorithm. Figure 1 is the Chi-Square and ACO system architecture for the FS problem. We tested this system for aspect prediction functions in research article documents. ArXiv is a free distribution service and an open-access archive of research papers. We conducted the study using the ArXiv research article dataset. This dataset comprises 20,972 research articles covering various topics, such as computer science, physics, mathematics, statistics, quantitative biology, and quantitative finance. Leveraging this extensive dataset aims to gain valuable insights and develop a powerful FS technique capable of effectively classifying research papers based on abstracts. We used the abstract as an independent variable to determine the appropriate topic for the article. The labels used in the dataset are computer science, physics, mathematics, statistics, biology, and quantitative economics.

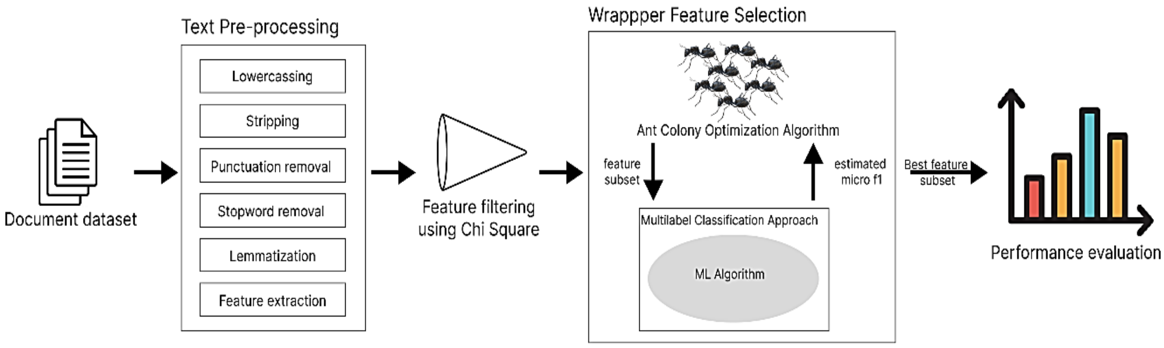

Figure 1. The system architecture of CSACO algorithm

We improve the performance of the multi-label classifier by a two-phase FS approach: Identify relevant features that can improve performance and identify a small set of features with minimum redundancy, which can reduce computational costs without reducing performance. A hybrid filter-wrapper FS approach is proposed to tackle these two objectives. The CS method is used to eliminate many irrelevant and redundant features. ACO is applied to reduce irrelevant and redundant features, as well as to improve classifier performance. Therefore, the proposed hybrid filter-wrapper approach leverages the filter method's efficiency and the wrapper method's accuracy to complement each approach's shortcomings.

3.1. Data preprocessing

Text preprocessing is the first step in natural language processing (NLP) tasks, as it helps to clean and prepare textual data for further analysis. In this research, the preprocessing consists of four stages: lowercasing and stripping, removing punctuation and stop-word removal, lemmatization, and feature extraction. i) Lowercasing and stripping: The first step in text preprocessing is to convert text to lowercase.

The lower casing ensures consistency by treating uppercase and lowercase versions of the same word as identical. Removing leading and trailing white spaces using the stripping technique enhances data cleanliness. Stripping is a simple but essential text preprocessing step that helps ensure consistency in the text data, including ii) Removing punctuation and stopwords: Removing punctuation marks from the text is necessary to extract meaningful information, simplify the data, and minimize unnecessary noise during analysis. The stopwords are commonly used words in the language that do not carry significant meaning and could be safely ignored during analysis. Removing stop words reduces the dimensionality of the text data, leading to more efficient NLP tasks; iii) lemmatization reduces words to their base or root form, known as lemmas. This step employs part-of-speech (POS) tagging to assign the appropriate POS tags to each word in the text. Lemmatization maps each word to its lemma based on the POS tags. This process helps reduce word variations and enables better semantic analysis; iv) feature extraction. Text extraction transforms text data into numerical features that could be used for machine learning or information retrieval tasks [28].

Bag of words (BOW) is a technique for text extraction that represents text documents as an extensive collection of words, ignoring grammar and word order, but still maintaining term frequency. In BOW, the text will be converted into vectors for each document by counting the occurrences of each word in the vocabulary according to the chosen measure-the preceding transformation results in a distribution where the values vary approximately from -1 to 1 [6].

$$x\ scaled = \frac{x}{\max(x)} \tag{1}$$

3.2. Feature selection

Although there are many classifiers for text categorization, the main challenge of text classification is the high dimensionality of the feature space. A document typically contains hundreds or thousands of words considered features, but many are possibly uninformative or redundant regarding class labels. Many scientists conducted thorough experiments to solve this problem. After this, these high-dimensional variables and data can be efficiently divided into essential aspects. The primary motivation of FS tasks is dimensionality minimization in large multidimensional datasets. FS innovation is a significant step in the success of knowledge discovery in a problem with many features. The FS process allows the elimination of attributes that can help determine data size, reduce computational time and requirements, minimize dimensionality, and improve performance predictors [5].

The literature classifies the FS process into filter, wrapper, embedded, and hybrid techniques. The filter model evaluates the features without utilizing ML algorithms, relying on data characteristics. The filter model considers the relevance of features without using any learning algorithm. This model will evaluate and rank features based on information theory measurements and select the feature with the highest ranking. The most common filter methods are information gain (IG), mutual information (MI), and CS. IG measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term [29]. However, MI measures the dependence between variable terms and categories. The CS statistical formula is related to the FS function of information theory [7].

In addition, the learning model involves a wrapper approach requiring high computation. Therefore, the selected attributes have an impact on increasing accuracy. The advantage of the wrapper technique is that it optimizes the classifier's performance. On the other hand, embedded FS methods are implemented using algorithms with their own built-in FS methods. Recognized examples of embedded methods are decision trees, least absolute shrinkage and selection operator (LASSO), and regression. Meanwhile, LASSO regression is a regularization technique used in regression methods for more accurate predictions [30].

Hybrid techniques use multiple FS strategies to create subsets. The technique combines several approaches to obtain the best feature subset rather than the independent methods. This hybrid approach combines two methods: wrapper and filter. We devised a two-step approach to streamline the feature selection process in the proposed architecture. The initial step entails leveraging the power of the CS test, the statistical measure of dependence between two categorical variables.

3.2.1. Chi-square

Chi-square (CS) is a data mining technique for implementing FS filters whose results are comparable to other strategies. CS selects features regarded as essential for classification and could remove parts that do not affect the target class [7]. Filter techniques calculate the relationship between existing features and goal categories to facilitate CS [31]. In the first stage, the CS method is applied to determine the importance of each feature as follows in (2) [31].

$$x^2(t_k, c) = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{2}$$

where N is the total number of training documents; A is the number of documents in c containing t; B is the number of documents not in c, containing t; C is the number of documents in c not containing t; and D is the number of documents not in c not containing t. Examining this data set, we can discern features that show significant relationships with the target variable. Consequently, this enables us to effectively reduce the dimensionality of the feature space and concentrate our efforts on the most promising and influential features.

3.2.2. Ant colony optimization

Ant colony optimization (ACO) is an algorithm in the SI group, a type of paradigm development used to solve optimization problems based on the behavior of ant swarms. In ACO, every time an ant moves, it will leave pheromones (a kind of chemical) on the path it passes. This pheromone is a signal to fellow ants. Routes that are heavily travelled will have a stronger signal. The following ants usually choose the path with the strongest signal to find the shortest path to the food source. This ACO algorithm is represented in graph form when solving FS problems. In this graph, features are depicted as nodes, and the edges connecting them indicate the sequence of selected features. ACO aims to find the optimal feature subset for guiding the ant's traversal through the graph. The ant seeks to visit the minimum number of nodes while meeting the traversal-stopping criterion [32], [33].

Figure 2 illustrates the ACO representation step in FS. Any feature can be chosen as the next option because nodes are fully connected. The ant currently resides at node f_1, where it can select the feature added to its journey (dotted lines) [32]. Following the transition rule, the ant proceeds with feature f_2, then f_3, and subsequently f_4. Once the ant reaches f_4, it confirms that the current subset {f_1, f_2, f_3, and f_4} fulfils the traversal stopping criteria.

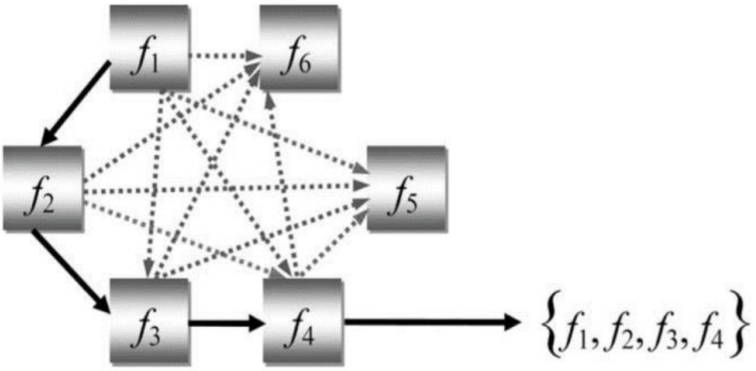

Figure 2. ACO Representation in feature selection

Consequently, this feature subset is a potential candidate for data compression. The graph representation can be reformed using standard ACO algorithm transitions and pheromone update rules [18]. The ant will move through the nodes, looking for the best subset of features if the stopping condition is unsatisfied. Heuristic desirability and pheromone levels are the two components that make up the probabilistic transition rule, as shown in (3) [17].

$$p_{ij}^{k}(t) = \begin{cases} \frac{[\tau_i(t)]^A[\eta_i]^B}{\sum_{j\in J^k}[\tau_j(t)]^A[\eta_j]^B}, & if\ i \in J^k \\ 0 & otherwise \end{cases} \tag{3}$$

where J^k is the set of ant-k's unvisited features, η_i is the heuristic desirability of element-i.

The $\tau_i(t)$ is the pheromone value at feature-i. While η_j is the heuristic value of element-j, and $\tau_j(t)$ is the pheromone value of feature-j. At the same time, A and B are parameters that determine the relative importance of the pheromone value and heuristic information. Pheromone evaporation and pheromone deposit are two steps in pheromone management. The basic procedure, the pheromone update, consists of both strategies. Pheromone evaporation prevents ants from following the same path and developing the same inclusion. The winning then had an advantage over the other ants by having the best answer, and all ants could update the pheromone level on the features they visited [34].

This research applies the ACO algorithm for the FS technique. We initialize it with reduced features and define a fitness function to evaluate the quality of feature subsets. The fitness function measures the performance of the MLC model using selected features. Fitness functions are essential for assessing the quality of feature subsets, including metrics such as accuracy, F1-score, or area under the receiver of the operating characteristic curve.

Our research used the micro F1-score as fitness value. The micro F1-score considered the overall precision and recall of the classification model, providing a comprehensive measure of its effectiveness in capturing all relevant labels across the dataset. We adopted the classifier chain (CC) approach to tackle the MLC task's multi-label nature. This approach allows us to handle multiple labels simultaneously, enhancing the accuracy and effectiveness of the process. CC is one of the conventional MLC methods based on the transformation technique problem. This method is a direct extension of binary relevance (BR), developed to address the label correlation problem. In BR, labels are considered independent classifiers until the algorithm ignores the correlation between labels.

On the contrary, in CC, the labels are the chain structure, which allows communication (i.e., sharing predictions) among the underlying classifiers. This approach usually determines the chain order based on the label's dependency on the data set. The CC approach can combine label dependencies, which allows CC to capture the correlation between labels, particularly not independent labels. Regardless, this approach may be more computationally expensive than the One vs Rest classifier since the predictions for each label depend on all the preceding labels on the chain [35].

This research aims to provide a detailed comparative analysis of the performance of three ML algorithms: MNB, CNB, and LSVC. MNB algorithm is a probabilistic learning method widely used in NLP. MNB is suitable for classification with discrete features. Multinomial distributions usually require an integer number of features. However, fractional counting, as in feature weighting, can also work in practice. This algorithm is based on Bayes' theorem and predicts text tags such as a piece of email or articles [36]. Subsequently, CNB adapts the standard MNB algorithm. MNB could perform better on imbalanced data sets. An unbalanced dataset is a data set where the example number of one class is greater than the number of instances that belong to another category. The sample distribution is not uniform. This dataset type can be challenging because the model can easily overfit this data in favor of classes with more examples. In addition, a linear SVM is used for linearly separable data. If a data set can be classified into two types using one straight line, later, the data is said to be linearly separable. The classifier is called a linear SVM (LSVC) classifier.

4. RESULT AND DISCUSSION

This research uses a dataset from the ArXiv research articles. This dataset comprises 20,972 document abstracts, each of which could cover more than one topic. In this study, we compared the performance of CS-ACO, CS-GWO, and CS only. For the CS method, we set the significance level threshold to 0.005, reducing features from 44,858 to 12,430. Meanwhile, in CS-ACO, the feature reductions listed in Table 1 are obtained.

In the case of ACO, we performed the training data for five epochs utilizing varying populations of ant agents, namely 25, 50, 100, and 150. After the FS stage, a classification process is carried out. In this research, the multi-label classifier is used to predict class labels. The performance evaluation of the proposed method in the CS-ACO feature selection process has primarily been automated. However, to gain deeper insights into the performance of the proposed method, we employed two additional metrics: hamming loss and macro F1-score. These metrics allowed us to assess the effectiveness and accuracy of the classification process, providing valuable information for further analysis. For example, accuracy is the correct label prediction over the total number of labels. Complete accuracy is average overall examples. For micro averaging, true positive (TP), false positive (FP), and false negative (FN) are all classes used to calculate micro average precision and micro average recall. The equations (4) to (7) depict the calculations of Recall, Precision, MacroF1, and MicroF1 [37].

$$Recall = \frac{TP}{TP+FP} \tag{4}$$

$$Precision = \frac{TP}{TP+FN} \tag{5}$$

$$F1_{macro} = \frac{\sum_{c_i \in C} F1score}{|C|} \tag{6}$$

$$F1_{micro} = \frac{\sum_{C=1}^{n} TP_C}{\sum_{C=1}^{N}(TP_C+\frac{1}{2}(FP_C+FN_C)} \quad (7)$$

In this research, the evaluation method uses Hamming loss (HL). This method evaluated the classifier based on the false prediction from all the prediction classes in (8). Where N is total documents, L is total class, $\hat{y}_j^{(i)}$ is prediction class, $\hat{y}_j^{(i)}$ is actual class, and $y_j^{(i)} \neq \hat{y}_j^{(i)}$ is false prediction overall prediction class, in this case, is difference between prediction class and actual class [38].

$$\frac{1}{nL}\sum_{i=1}^{n}\sum_{j=1}^{L}[(y_j^{(i)} \neq \hat{y}_j^{(i)})] \quad (8)$$

Table 1 shows the performance of CS-ACO with the same algorithms and approach, MNB, CNB, and LSVC, on different populations of agents. The parameters have been adjusted: the pheromone coefficient is 0.2, and the evaporation rate is 0.8. CS-ACO with the MNB algorithm works best among the three other algorithms. Results are visible in the 150 population: the MicroF1 score is 80%, and the HL score is 0.0868.

Table 1. The CSACO approach of MNB, CNB, and LSVC

Population	Multinomial naive Bayes				Complement naive Bayes				Linear SVC			
	Micro F1	Macro F1	HL	Features	Micro F1	Macro F1	HL	Features	Micro F1	Macro F1	HL	Features
25	0.7977	0.659	0.0878	**10946**	0.7916	**0.7054**	0.0963	**10997**	0.7772	0.638	0.0955	10581
50	0.7987	0.6664	0.0877	10894	0.792	0.6915	0.0957	10926	0.7801	0.6304	0.094	**10647**
100	0.7999	**0.6691**	0.087	10927	**0.7932**	0.7022	**0.0954**	10941	0.7784	0.6315	0.0946	9868
150	**0.8**	0.6657	**0.0868**	10609	0.792	0.7017	0.0958	10578	**0.7847**	**0.6634**	**0.092**	9922

Table 2 shows the comparative performance of MNB, CNB, and LSVC in the CC approach on different agent populations of grey wolves run over five periods. This experiment shows that increasing the GWO agent population increases system performance. These experiments are seen in the F1 micro score in the MNB and CNB classifiers. The MNB classifier algorithm works best compared to other algorithms when the GWO population is 150, the value of MicroF1 is 79.75%, and the HL value is 0.0878. Table 3 displays experiments using the CS algorithm. In Table 3, MNB has the highest performance, with a micro F1-score of 79.64% and a HL score of 0.089. Table 4 describes the algorithm without FS.

Table 2. The CSGWO approach of MNB, CNB, and LSVC

Population	Multinomial naive Bayes				Complement naive Bayes				Linear SVC			
	Micro F1	Macro F1	HL	Features	Micro F1	Macro F1	HL	Features	Micro F1	Macro F1	HL	Features
25	0.794	0.6458	0.0891	9746	0.7865	0.6722	**0.0984**	9180	**0.7904**	0.6362	0.0901	**9226**
50	0.7938	**0.6629**	0.0897	**11080**	0.7885	0.6803	0.0972	10058	0.7757	**0.6433**	0.0968	6125
100	0.7939	0.6453	0.0895	10947	0.7897	**0.6845**	0.097	**10403**	0.7718	0.6191	0.0985	6249
150	**0.7975**	0.639	**0.0878**	10514	**0.7925**	0.6584	0.0894	9660	0.7719	0.623	**0.0989**	6136

Table 3. CS algorithm only

Algorithm	Micro F1	Macro F1	Hamming loss
CC multinomial naive Bayes	**0.7964**	0.6685	**0.089**
CC complement naive Bayes	0.7894	**0.6988**	0.0972
CC linear SVC	0.7717	0.6134	0.0972

Table 4. Without feature selection

Algorithm	Micro F1	Macro F1	Hamming loss
CC multinomial naive Bayes	0.7778	**0.546**	**0.0928**
CC complement naive Bayes	**0.7804**	0.5576	0.0969
CC linear SVC	0.7717	0.6419	0.0961

5. CONCLUSION

The research contributes to FS methodology, which harnesses the combined power of the chi-square and ACO algorithms. The method demonstrates its efficiency through rigorous experimentation and analysis in achieving higher F1-measure and low HL values while utilizing a significantly diminished set of features. This outcome substantiates the effectiveness of the proposed approach in enhancing the performance of FS tasks. The text classifier uses MNB with an agent population of 150-comparisons on CS, without FS, and CS-GWO. In CS, the result was a micro F1-score of 79.64%. Meanwhile, the result without FS is 78.04%. Subsequently, in CS-GWO, the results show that the micro F1 value is 79.75%, and HL is 0.0878.

The MNB algorithm worked best in a population of 150 agents, with a micro F1-score of 80% and a HL score of 0.0868. Thus, the results are more optimal using the CS-ACO FS method and MNB classifier on 150 populations of 80%. However, optimizing FS techniques is still an ongoing effort, and many opportunities exist for future exploration and improvement. This research also thoroughly investigates other FS methodologies to validate further and refine the proposed method's advantages. Additionally, considering the tremendous progress in machine learning models, exploring the performance of such models alongside the proposed techniques would be beneficial. Such analyses can reveal new insights and enrich our understanding of the interactions between machine learning models and swarm intelligence methods. Therefore, in future efforts, we will explore incorporating hybrid FS methods, which combine the strengths of multiple swarm Intelligence algorithms, resulting in potential breakthroughs in performance and interpretability in the multi-label classification of text documents.

ACKNOWLEDGEMENTS

The Grants from The Ministry of Education, Culture, Research, and Technology of Indonesia supported this work, PDD Grant 1900/UN1/DITLIT/Dit-Lit/PT.01.03/2022.

REFERENCES

[1] Z. Chen and J. Ren, "Multi-label text classification with latent word-wise label information," *Applied Intelligence*, vol. 51, no. 2, pp. 966–979, 2021.

[2] X. Yan and J. Bien, "Rare feature selection in high dimensions," *Journal of the American Statistical Association*, vol. 116, no. 534, pp. 887–900, 2021, doi: 10.1080/01621459.2020.1796677.

[3] X. Zhou *et al.*, "A survey on text classification and its applications," *Web Intelligence*, vol. 18, no. 3, pp. 205–216, 2020, doi: 10.3233/WEB-200442.

[4] J. Ma, B. Xue, and M. Zhang, "A hybrid filter-wrapper feature selection approach for authorship attribution," *International Journal of Innovative Computing, Information and Control*, vol. 15, no. 5, pp. 1989–2006, 2019.

[5] L. Brezočnik, I. Fister, and V. Podgorelec, "Swarm intelligence algorithms for feature selection: a review," *Applied Sciences (Switzerland)*, vol. 8, no. 9, 2018, doi: 10.3390/app8091521.

[6] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: a review," *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 3797–3816, 2019, doi: 10.1007/s11042-018-6083-5.

[7] S. Bahassine, A. Madani, M. Al-Sarem, and M. Kissi, "Feature selection using an improved chi-square for Arabic text classification," *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 2, pp. 225–231, Feb. 2020, doi: 10.1016/j.jksuci.2018.05.010.

[8] O. M. Alyasiri, Y.-N. Cheah, A. K. Abasi, and O. M. Al-Janabi, "Wrapper and hybrid feature selection methods using metaheuristic algorithms for English text classification: a systematic review," *IEEE Access*, vol. 10, pp. 39833–39852, 2022, doi: 10.1109/ACCESS.2022.3165814.

[9] J. Yousef, A. Youssef, and A. Keshk, "A hybrid swarm intelligence based feature selection algorithm for high dimensional datasets," *IJCI International Journal of Computers and Information*, 2021, doi: 10.21608/ijci.2021.62499.1040.

[10] L. Kumar and K. K. Bharti, "A novel hybrid BPSO–SCA approach for feature selection," *Natural Computing*, vol. 20, no. 1, pp. 39–61, 2021, doi: 10.1007/s11047-019-09769-z.

[11] J. Y. Khaseeb, A. Keshk, and A. Youssef, "A hybrid swarm intelligence feature selection approach based on time-varying transition parameter," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 13, no. 1, pp. 781–795, 2023, doi: 10.11591/ijece.v13i1.pp781-795.

[12] M. Ghosh, R. Guha, I. Alam, P. Lohariwal, D. Jalan, and R. Sarkar, "Binary genetic swarm optimization: a combination of GA and PSO for feature selection," *Journal of Intelligent Systems*, vol. 29, no. 1, pp. 1598–1610, 2020, doi: 10.1515/jisys-2019-0062.

[13] R. Sharma, P. Marikkannu, and A. Sungheetha, "Three-dimensional MRI brain tumour classification using hybrid ant colony optimisation and grey wolf optimiser with proximal support vector machine," *International Journal of Biomedical Engineering and Technology*, vol. 29, no. 1, 2019, doi: 10.1504/IJBET.2019.096879.

[14] A. Elnawasany, M. A. A. Makhlouf, B. Tawfik, and H. Nassar, "Feature selection of unbalanced breast cancer data using particle swarm optimization," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 5, pp. 4951–4959, Oct. 2022, doi: 10.11591/ijece.v12i5.pp4951-4959.

[15] H. Xu, X. Liu, and J. Su, "An improved grey wolf optimizer algorithm integrated with Cuckoo search," in *2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, Sep. 2017, vol. 1, pp. 490–493, doi: 10.1109/IDAACS.2017.8095129.

[16] D. Emmanuel, S. Joseph, D. Oyewola, A. A. Fadele, and H. C. Muhammad, "Application of grey wolf optimization algorithm: recent trends, issues, and possible horizons," *Gazi University Journal of Science*, vol. 35, no. 2, pp. 485–504, Jun. 2022, doi: 10.35378/gujs.820885.

[17] K. Menghour and L. Souici-Meslati, “Hybrid ACO-PSO based approaches for feature selection,” *International Journal of Intelligent Engineering and Systems*, vol. 9, no. 3, pp. 65–79, 2016.

[18] M. Ghosh, R. Guha, R. Sarkar, and A. Abraham, “A wrapper-filter feature selection technique based on ant colony optimization,” *Neural Computing and Applications*, vol. 32, no. 12, pp. 7839–7857, Jun. 2020, doi: 10.1007/s00521-019-04171-3.

[19] Z. Wang, S. Gao, M. Zhou, S. Sato, J. Cheng, and J. Wang, “Information-theory-based nondominated sorting ant colony optimization for multiobjective feature selection in classification,” *IEEE Transactions on Cybernetics*, vol. 53, no. 8, pp. 5276–5289, Aug. 2023, doi: 10.1109/TCYB.2022.3185554.

[20] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey wolf optimizer,” *Advances in Engineering Software*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[21] M. B. Subkhi, C. Fatichah, and A. Z. Arifin, “Feature selection using hybrid binary grey wolf optimizer for Arabic text classification,” *IPTEK The Journal for Technology and Science*, vol. 33, no. 2, Art. no. 105, Sep. 2022, doi: 10.12962/j20882033.v33i2.13769.

[22] R. Purushothaman, S. P. Rajagopalan, and G. Dhandapani, “Hybridizing gray wolf optimization (GWO) with grasshopper optimization algorithm (GOA) for text feature selection and clustering,” *Applied Soft Computing*, vol. 96, Nov. 2020, doi: 10.1016/j.asoc.2020.106651.

[23] C. Shen and K. Zhang, “Two-stage improved Grey Wolf optimization algorithm for feature selection on high-dimensional classification,” *Complex and Intelligent Systems*, Jul. 2021, doi: 10.1007/s40747-021-00452-4.

[24] H. Chantar, M. Mafarja, H. Alsawalqah, A. A. Heidari, I. Aljarah, and H. Faris, “Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification,” *Neural Computing and Applications*, vol. 32, no. 16, pp. 12201–12220, Aug. 2020, doi: 10.1007/s00521-019-04368-6.

[25] O. S. Qasim and Z. Y. Algamal, “A gray wolf algorithm for feature and parameter selection of support vector classification,” *International Journal of Computing Science and Mathematics*, vol. 13, no. 1, p. 93, 2021, doi: 10.1504/IJCSM.2021.114185.

[26] S. Tabakhi and P. Moradi, “Relevance–redundancy feature selection based on ant colony optimization,” *Pattern Recognition*, vol. 48, no. 9, pp. 2798–2811, Sep. 2015, doi: 10.1016/j.patcog.2015.03.020.

[27] Q. Al-Tashi, S. J. Abdul Kadir, H. M. Rais, S. Mirjalili, and H. Alhussian, “Binary optimization using hybrid grey wolf optimization for feature selection,” *IEEE Access*, vol. 7, pp. 39496–39508, 2019, doi: 10.1109/ACCESS.2019.2906757.

[28] A. Bhavani and B. Santhosh Kumar, “A review of state art of text classification algorithms,” in *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, Apr. 2021, pp. 1484–1490, doi: 10.1109/ICCMC51019.2021.9418262.

[29] M. Mojaveriyan, H. Ebrahimpour-komleh, and S. jalaleddin Mousavirad, “IGICA: A hybrid feature selection approach in text categorization,” *International Journal of Intelligent Systems and Applications*, vol. 8, no. 3, pp. 42–47, 2016, doi: 10.5815/ijisa.2016.03.05.

[30] E. O. Abiodun, A. Alabdulatif, O. I. Abiodun, M. Alawida, A. Alabdulatif, and R. S. Alkhawaldeh, “A systematic review of emerging feature selection optimization methods for optimal text classification: the present state and prospective opportunities,” *Neural Computing and Applications*, vol. 33, no. 22, pp. 15091–15118, Nov. 2021, doi: 10.1007/s00521-021-06406-8.

[31] W. BinSaeedan and S. Alramlawi, “CS-BPSO: Hybrid feature selection based on chi-square and binary PSO algorithm for Arabic email authorship analysis,” *Knowledge-Based Systems*, vol. 227, Sep. 2021, doi: 10.1016/j.knosys.2021.107224.

[32] J. A. Widians, R. Wardoyo, and S. Hartati, “A study on text feature selection using ant colony and grey wolf optimization,” in *2022 Seventh International Conference on Informatics and Computing (ICIC)*, Dec. 2022, pp. 1–7, doi: 10.1109/ICIC56845.2022.10007019.

[33] M. F. F. Ab Rashid, “A hybrid ant-wolf algorithm to optimize assembly sequence planning problem,” *Assembly Automation*, vol. 37, no. 2, pp. 238–248, Apr. 2017, doi: 10.1108/AA-11-2016-143.

[34] M. Paniri, M. B. Dowlatshahi, and H. Nezamabadi-pour, “MLACO: a multi-label feature selection algorithm based on ant colony optimization,” *Knowledge-Based Systems*, vol. 192, Art. no. 105285, Mar. 2020, doi: 10.1016/j.knosys.2019.105285.

[35] A. Abdullahi, N. A. Samsudin, S. K. A. Khalid, and Z. A. Othman, “An improved multi-label classifier chain method for automated text classification,” *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 3, 2021.

[36] S. Xu, Y. Li, and Z. Wang, “Bayesian multinomial naïve Bayes classifier to text classification,” in *Advanced Multimedia and Ubiquitous Engineering: MUE/FutureTech 2017 11*, Springer, 2017, pp. 347–352, doi: 10.1007/978-981-10-5041-1_57.

[37] M. Heydarian, T. E. Doyle, and R. Samavi, “MLCM: multi-label confusion Matrix,” *IEEE Access*, vol. 10, pp. 19083–19095, 2022, doi: 10.1109/ACCESS.2022.3151048.

[38] S. Destercke, “Multilabel prediction with probability sets: the hamming loss case,” in *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Springer, 2014, pp. 496–505, doi: 10.1007/978-3-319-08855-6_50.

BIOGRAPHIES OF AUTHORS

Joan Angelina Widians is a doctoral student in the Department of Computer Science and Electronics at Gadjah Mada University, Yogyakarta, Indonesia. She received her master’s degree from the Department of Computer Science at Gadjah Mada University, Yogyakarta, Indonesia (2008). She received a bachelor’s degree in information systems from STMIK Widya Cipta Dharma, Samarinda, Indonesia (2000). She is a lecturer in the Informatics Department at Mulawarman University, Indonesia. Her research interests include expert systems, data mining, and swarm intelligence. She can be contacted at email: angelwidians@unmul.ac.id.

Retantyo Wardoyo is a professor and a researcher at the Department of Computer Science and Electronics, Gadjah Mada University, Indonesia. He obtained his bachelor's degree in mathematics at Gadjah Mada University Indonesia (1982). He obtained his master’s degree in computer science from the University of Manchester, UK (1990) and his Ph.D. in computation at the University of Manchester Institute of Science and Technology, UK (1996). His research interests include intelligent systems, reasoning systems, expert systems, fuzzy systems, group DSS and clinical DSS, and computational intelligence. He can be contacted at email: rw@ugm.ac.id.

Sri Hartati is a professor and researcher at the Department of Computer Science and Electronics, Gadjah Mada University, Indonesia. She received a bachelor of Electronics and Instrumentation from the Faculty of Mathematics and Natural Science, Gadjah Mada University, Indonesia (1986), received a Master of Science in computer science from the Faculty of Computer Science, University of New Brunswick, Canada (1990), and received Ph.D. of Computer Science from Faculty of Computer Science, University of New Brunswick, Canada (1996). Her research interests are in intelligent systems, knowledge-based systems, reasoning systems, expert systems, DSS, Clinical DSS, GDSS, medical computing, and computational intelligence. She can be contacted at email: shartati@ugm.ac.id.