Journal of Computer Science

Original Research Paper

Mining Sports Articles using Cuckoo Search and Tabu Search with SMOTE Preprocessing Technique

Waheeda Almayyan

CIS, Public Authority for Applied Education and Training (PAAET), Kuwait

Article history
Received: 05-01-2021
Revised: 09-03-2021
Accepted: 11-03-2021

Email: w_almayyan@yahoo.com

Abstract Sentiment analysis is one of the most popular domains for natural language text classification, crucial for improving information extraction. However, massive data availability is one of the biggest problems for opinion mining due to accuracy considerations. Selecting high discriminative features from an opinion mining database is still an ongoing research topic. This study presents a two-stage heuristic feature selection method to classify sports articles using Tabu search and Cuckoo search via Lévy flight. Lévy flight is used to prevent the solution from being trapped at local optima. Comparative results on a benchmark dataset prove that our method shows significant improvements in the overall accuracy from 82.6% up to 89.5%.

Keywords Sentiment Analysis, Subjectivity Analysis, Feature Reduction, Tabu Search, Cuckoo Search, Random Forest Classifier

Introduction

The Internet is a rich source of various points of view and an increasing number of individuals are using the Web as a medium for sharing their opinions and attitudes in text. This includes online product or service reviews, travel advice, social media discussions and blogs, customer recommendations, movie and book reviews and stock market predictions (Zhou *et al*., 2013). Therefore, this motivates providing tools to automatically extract and analyze public opinion for business marketing or social studies and understand consumers' preferences (Liu, 2010).

Sentiment analysis, which involves evaluating sentences as objective or subjective, is challenging to interpret natural language as subjectivity needs more investigation (Pang and Lee, 2008). Moreover, subjectivity analysis depends on the fact that expressions and sentence phrases may express varying intensities depending on the context in which they occur. Furthermore, articles of text need not be entirely classified as subjective or objective. Hence, subjectivity can be expressed in different ways as proposed in (Liu, 2012) and overall, it is considered highly domain-dependent since it is affected by the sentiments of words.

There is a great need to develop an automated solution to differentiate objective and subjective articles (Pang and Lee, 2008). Consequently, many features have been reported to select the best presentation for subjectivity detection, ranging from the lexicon and syntactic features to semantic features, including phrase pattern, N-grams, character and word-level lexical features phrase-level sentiment scores (Liu, 2012). As a result, the large scale of such feature datasets is a significant challenge.

Feature selection aims to significantly minimize the computational overhead and consequently enhance the overall classification performance through eliminating irrelevant and insignificant features from datasets before model implementation (Pang *et al*., 2002; Turney, 2002), which is an essential requirement in text-based sentiment-analysis problems.

Numerous machine learning algorithms have been proposed for sentiment analysis or opinion mining and related feature selection, including k-nearest neighbor methods (Wiebe *et al*., 2002), bootstrapping algorithms (Wiebe and Riloff, 2011), genetic algorithm approaches (Rizk and Awad, 2012), SVM model, approaches (Heerschop *et al*., 2011; Abbasi *et al*., 2011) and naïve Bayes classifiers (Yu and Hatzivassiloglou, 2003).

The Tabu search and Cuckoo search algorithms have gained significant attention from researchers (Chen *et al*., 2021; Srivastava *et al*., 2012). The proposed work's motivation is to design a two-stage bio-inspired hybrid algorithm based on Tabu search and Cuckoo search via Lévy flight to select the optimal features in a sports-related text dataset for subjectivity analysis. This study combines the Tabu algorithm's ability to converge to a solution and the Cuckoo mechanism of backtracking from local optima by Lévy flight (Glover, 1989). Cuckoo search has been widely used in adaptive search strategies for constructing computational models (Yang and Deb, 2009). One of the most desirable features of the algorithm is computationally efficient and easy to implement with a smaller number of parameters (Yang and Deb, 2009). Tabu Search (TS) is added to reduce the number of iterations and execution time of the algorithm, thus reducing the overall complexity (Chen *et al*., 2021). Among machine learning algorithms, the Random Forest method (RandF) has received increased attention within several classification problems (Glover, 1989; Yang and Deb, 2009). RandF is an ensemble machine learning technique that was developed by (Breiman, 2001). This classifier has been well utilized in many classification problems but is relatively uncommon in sentiment analysis.

The main objective is to improve the prediction accuracy with a resampling scheme to overcome the dataset imbalance issue as there is a tradeoff between the accuracy and size of the generated feature subsets. In addition to the methods mentioned above, techniques in this study, MLP, SimpleLogistic, k-NN, RandF and C4.5 classifiers will be used to compare the performance of our proposed feature selection technique in terms of precision, ROC and Cohen's kappa coefficient using the dataset used by (Hajj *et al*., 2019).

The main contributions of this article are as follows:

- Consider the sentiment analysis problem to having two stages
- Apply Tabu search and Cuckoo search via Lévy flight to feature selection
- Apply SMOTE technique to balance the training data in the classification stage
- Apply several classification models in the classification stage

The remainder of this study is organized in the following manner. Section 2 explains the theoretical approach of feature selection methods and the proposed technique. The evaluation procedure, the dataset and the experimental results are presented in section 3. Finally, the conclusions are summarized in section 4.

Methodology

One of the motivation goals in research to improve classification performance is applying hybrid-learning approaches instead of individual ones. We first select features by the Cuckoo search algorithm in our method and then we apply Tabu search to construct a new feature subset.

Tabu Search Technique

Tabu search is an iterative memory-based algorithm that Glover proposed in 1986 to solve combinatorial optimization problems (Glover, 1989; 1990). Since then, Tabu search has been successfully applied in several multiclass classification problems (Hajj *et al*., 2019). It comprises of local search mechanism combined with Tabu mechanism.

Tabu search starts with an initial solution $X' \in \Omega$ among neighborhood solutions, where Ω is the set of feasible solutions. Then the algorithm searches and evaluate all the possible neighbor's solution $N(X) \in \Omega$ to obtain a new one with an improved functional value. A solution candidate $X' \in \Omega$ $N(X)$ can be reached from X if the new solution X' is not registered in the Tabu list, or it satisfies the aspiration criterion (Tahir *et al*., 2004b). If the candidate solution X' is better than X_{best}, the value of Xbest is overridden; otherwise Tabu search will go uphill to avoid local minima.

Tabu search avoids cycling by limiting visiting previously visited solutions for a certain number of iterations. This undoubtedly improves the performance of the local search. Then, the neighborhood search resumes based on the new solution X' until the stopping criterion is met (Korycinski *et al*., 2003; Sait and Youssef, 1999).

Cuckoo Search Algorithm

The Cuckoo search algorithm is basically derived from the strange reproductive behavior of particular cuckoo species. These species choose to put eggs in randomly chosen nests of other host birds but have similar matching patterns of the hosts' own eggs to reduce their ability to discover them (Yang and Deb, 2014). The cuckoos rely on these host birds to accommodate their eggs. Sometimes, when the host birds recognize unfamiliar eggs, it usually rejects it or abandons their nests.

According to the cuckoo algorithm, each egg in the nest represents a possible solution and the foreign cuckoo egg represents a new solution. The goal is to employ potentially better solutions (cuckoos) to replace the nests' solution (Civicioglu and Besdok, 2013). Cuckoo search algorithm generates a new candidate solution (nest) $x_i^{(r+1)}$ for a cuckoo *n* (Kaveh and Bakhshpoori, 2013; Rodrigues *et al*., 2013):

$$x_i^{(r+1)} = x_i^{(r)} + \alpha \otimes Levy(\lambda) \tag{1}$$

where, *s* is the step size and *a* >0 is the step size scaling, which is related to the problem of interest. In most cases, α is set to 1. The symbol ⊕ is an entry-wise multiplication that is similar to those used in the PSO algorithm.

The Cuckoo search is based on Lévy flights to avoid a local optimum (Korycinski *et al*., 2003). The concept of Lévy flights explores the solution space (s) by providing a random walk with random steps drawn from a Lévy distribution for large steps (Tahir *et al*., 2004a), given by:

$$Levy(\lambda) \sim u = s^{-\lambda}, (1 < \lambda \le 3) \tag{2}$$

Synthetic Minority Over-Sampling (SMOTE) Technique

Through the classification process, especially in imbalanced datasets, there is a great challenge where classifiers tend to ignore minority classes. SMOTE technique was suggested by Chawla *et al*. in 2002 as a solution is to under-sample the majority class towards improving classification sensitivity in the minority class instances. The SMOTE technique adopts an over-sampling approach that operates at the feature level to balance the number of instances (Chawla *et al*., 2002).

The SMOTE technique works by resampling minority class through randomly interpolating new synthetic instances between the minority class and its nearest neighbours based on over-sampling rate, β% and the number of the nearest minority class data samples neighbours (*K*). Depending on the required rate β%, the SMOTE randomly adds new instances until the dataset is balanced. For illustration, for a minority data sample (x_o) if the required balance rate β% is 200% and samples neighbours *K* is 3, one of every three nearest neighbors is randomly repeated two times. A line is created using a random *K*th neighbor linking x_o to this neighbor and then, a random point on the line is selected to create one synthetic instance. Thus, any new synthetic instance x_s is created by:

$$x_s = x_o + \delta.\left(x_o^{\{t\}} - x_o\right) \tag{3}$$

where, x_s denotes a new synthetic instance, $x_o^{\{t\}}$ is the *t*th selected nearest neighbor of x_o in the minority class and δ is a random number ($\delta \in [0,1]$).

Random Forest Classifier (RandF)

Breiman (1996) proposed a new and promising tree-based ensemble classifier called RandF, which is based on a combination tree of predictors. RandF consists of a combination of individual base classifiers where each tree is generated using a random vector sampled independently from the classification input vector to enable a much faster construction of trees. For classification, all trees' classification votes are combined using a rule-based approach or based on an iterative error minimization technique by reducing the weights for the correctly classified samples.

The building of an ensemble of classifiers in RandF can be summarized as follows (Breiman, 1996):

- The RandF training algorithm starts with constructing multiple trees:
    - In this study, we use the random trees in building the RandF classifier with no pruning, which makes it light from a computational perspective
- The next step is preparing the training set for each tree, which is formed by randomly sampling the training dataset using a bootstrapping technique with replacement:
    - This step is called the bagging step (Breiman, 2001). The selected samples are called in-bag samples and the rest are set aside as out-of-bag samples
    - For each new training set generated, approximately one-third of the in-bag set data are duplicated (sampling with replacement) and used for building the tree. The remaining training samples (out-of-bag samples) are used to test the tree classification performance. Figure 1 illustrates the data sampling procedure. Each tree is constructed using a different bootstrap sample
- RandF increases the trees' diversity by choosing and using a random number of features (four features in this study) to construct the nodes and leaves of a random tree classifier. According to (Breiman, 2001), this step minimizes the correlation among the features, decreases the sensitivity to noise in the data and increases the accuracy of classification at the same time
- Building a random tree begins at the top of the tree with the in-bag dataset:
    - The first step involves selecting a feature at the root node and then splitting the training data into subsets for every possible value of the feature. This makes a branch for each possible value of the attribute. Tree design requires choosing a suitable attribute selection measure for splitting and the selection of the root node to maximize dissimilarity between classes
- If the information gain is positive, the node is split. Otherwise, the node becomes a leaf node that provides a decision for the most common target class in the training subset
- The partitioning procedure is repeated recursively at each branch node using the subset that reaches the branch and the remaining attributes, which continues until all attributes are selected
    - The highest information gain of the remaining attributes is selected as the next attribute. Eventually, the most occurring target class in the training subset that reached that node is assigned as the classification decision
- The procedure is repeated to build all trees
- After building all trees, the out-of-bag dataset is used to test trees and the entire forest. The obtained average misclassification error can be used to adjust the weights of the vote of each tree
- In this study, the implementation of RandF gives each tree the same weight

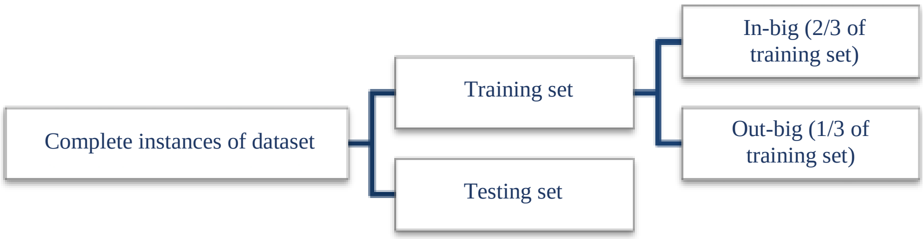
Complete instances of dataset
Training set
Testing set
In-big (2/3 of training set)
Out-big (1/3 of training set)

Fig. 1: Data partition in constructing random forest trees

Dataset and Experimental Results

Dataset

In this study, we used a dataset from a previous study (Hajj *et al*., 2019) for the assessment. The dataset comprises 52 features extracted from a corpus composed of 1,000 articles gathered from 658 sports articles, which were collected from over 50 interactive websites, including NBA.com, Fox Sports and Eurosport UK. The first 48 features are about the corpus's syntactic information, while the last 4 are semantic features.

The feature set is built over the concept of measuring the frequency counts of objective and subjective words in the text. Initially, it starts with summing positive, negative and objective scores and using it to normalize each word's text scores. Then, update the subjective and objective word counters according to a comparison between normalized scores and a threshold. Accordingly, the subjective word counter would be incremented if a word has a positive or negative score more significant than this threshold. Otherwise, the objective word counter would be incremented.

The list of features with their descriptions are shown in Table 1. A detailed explanation of the feature list can be found in a previous study (Rizk and Awad, 2012). Table 2 describes the class distribution over the two classes, which clearly shows that the dataset is imbalanced (65.8% instances are classified as objective statements).

Evaluation Metrics

The proposed model's performance is measured using precision, the Receiver Operator Characteristic (ROC) and Cohen's kappa coefficient. According to the confusion matrix in Table 3, precision is defined as:

$$precision = \frac{TN}{FP + TN} \times 100\% \tag{4}$$

The area under the ROC curve is a graphical plot for evaluating two-class decision problems. The ROC curve standard metric for analyzing classifier performance over a range of tradeoffs between True Positive (TP) and False Positive (FP) error rates (Dietterich, 2000; Smialowski *et al*., 2012). ROC usually ranges from 0.5 for an entirely random classifier and 1.0 for the perfect classifier.

Kappa error or Cohen's kappa coefficient is a useful measure to compare different classifiers' performance and the quality of selected features and ranges from 1 to -1 (Ben-David, 2008). When the kappa value approaches 1, there is a better chance for an agreement and When the kappa value approaches -1, it shows a low chance for agreement. The kappa error measure can be calculated using the following formula

$$Kappa\,error = \frac{P(A) - P(E)}{1 - P(E)} \tag{5}$$

$P(A)$ is the total agreement probability and $P(E)$ is the chance agreement's theoretical probability.

Results

Initially, feature selection is carried out in two main steps. Firstly, we construct a new reductive feature space using the Cuckoo search technique. In the first step, the original feature dimension is decreased from *a* to *b*. In the second step, the feature-length is reduced from *b* to *c*, the new feature space.

The results of the classifications with and without feature selection are reported. Table 4 reports the precision, ROC and Cohen's kappa coefficient of MLP, SimpleLogistic, k-NN, RandF and C4.5, to demonstrate the suggested feature selection techniques' performance which all use tenfold CV procedure evaluation (Schumacher *et al*., 1997). Prior to performing feature selection, the SimpleLogistic classifier (82.6%) slightly outperformed the RandF (82.1%), MLP (80.7%) and k-NN (75.6%) classifiers.

In the first phase, we investigated the effect of several feature set construction methods on classification performance. This task was carried out using four nature-inspired algorithms or swarm intelligence techniques: The Ant, Bat, PSO and Cuckoo algorithms. The selected features according to these techniques are summarized in Table 5. As noted, the dimensionality of sports article features was remarkably reduced. We reduced the size of the dataset from 52 attributes to only 17 to 23 attributes. For example, PSO helped in reducing the feature size by 67%.

Table 1: List of sports articles dataset features

Index	Feature	Feature description
1.	semanticobjscore	Number of coordinating and correlative conjunctions
2.	semanticsubjscore	Number of numerals and commonest cardinals
3.	CC	Number of determiners
4.	CD	Number of existential there
5.	DT	Number of foreign words
6.	EX	Number of subordinating preposition or conjunction
7.	FW	Number of ordinal adjectives or numerals
8.	INs	Number of comparative adjectives
9.	JJ	Number of superlative adjectives
10.	JJR	Number of list item markers
11.	JJS	Number of modal auxiliaries
12.	LS	Number of singular common nouns
13.	MD	Number of singular proper nouns
14.	NN	Number of plural proper nouns
15.	NNP	Number of plural common nouns
16.	NNPS	Number of pre-determiners
17.	NNS	Number of genitive markers
18.	PDT	Number of personal pronouns
19.	POS	Number of possessive pronouns
20.	PRP	Number of adverbs
21.	PRP$	Number of comparative adverbs
22.	RB	Number of superlative adverbs
23.	RBR	Number of particles
24.	RBS	Number of symbols
25.	RP	Number of "to" as preposition or infinitive marker
26.	SYM	Number of interjections
27.	TOs	Number of base form verbs
28.	UH	Number of past tense verbs
29.	VB	Number of present participle or gerund verbs
30.	VBD	Number of past participle verbs
31.	VBG	Number of present tense verbs with plural 3rd person subjects
32.	VBN	Number of present tense verbs with singular 3rd person subjects
33.	VBP	Number of WH-determiners
34.	VBZ	Number of WH-pronouns
35.	WDT	Number of possessive WH-pronouns
36.	WP	Number of WH-adverbs
37.	WP$	Number of quotation pairs in the entire article
38.	WRB	Number of questions marks in the entire article
39.	baseform	Number of exclamation marks in the entire article
40.	Quotes	Number of first-person pronouns (personal and possessive)
41.	Questionmarks	Number of second person pronouns (personal and possessive)
42.	Exclamationmarks	Number of third person pronouns (personal and possessive)
43.	pronouns1st	Number of comparative and superlative adjectives and adverbs
44.	pronouns2nd	Number of past tense verbs with 1st and 2nd person pronouns
45.	pronouns3rd	Number of imperative verbs
46.	compsupadjadv	Number of infinitive verbs (base form verbs preceded by “to”)
47.	past	Number of present tense verbs with 3rd person pronouns
48.	imperative	Number of present tense verbs with 1st and 2nd person pronouns
49.	present3rd	Number of words with an objective SENTIWORDNET score
50.	present1st2nd	Number of words with a subjective SENTIWORDNET score
51.	sentence1st	First sentence class
52.	sentencelast	Last sentence class

Table 2: Class distribution of the *sport articles* dataset

Index	Class label	Class size	Class distribution (%)
1	Objective	658	65.8
2	Subjective	342	34.2

Table 3: The confusion matrix

	Predicted class	
Hypothesis	Classified as objective	Classified as subjective
Positive	TP	FN
Negative	FP	TN

Table 4: Classification results before feature selection

Classifier	Performance index	Original data
MLP	Precision	0.807
	ROC	0.842
	Kappa error	0.580
SimpleLogistic	Precision	0.826
	ROC	0.869
	Kappa error	0.614
k-NN	Precision	0.756
	ROC	0.733
	Kappa error	0.473
RandF	Precision	0.821
	ROC	0.881
	Kappa error	0.612
C4.5	Precision	0.765
	ROC	0.714
	Kappa error	0.491

Table 5: Selected features of sports article dataset (1st stage)

FS technique	Number of selected features	Selected features
1. Ant	23	DT, EX, JJS, LS, NNPS, NNS, PDT, PRP$, RB, RBR, TOs, UH, VBN, VBP, VBZ, WDT, baseform, Quotes, questionmarks, pronouns2nd, compsupadjadv, imperative, present3rd
2. Bat	19	Semanticobjscore, semanticsubjscore, DT, EX, JJS, LS, NNPS, PRP$, RB, RBR, Tos, VBP, WDT, Quotes, questionmarks, exclamationmarks, imperative, present3rd, present1st2nd, Quotes, questionmarks, exclamationmarks, imperative, present3rd, present1st2nd
3. PSO	17	CD, DT, JJS, LS, NNPS, PRP$, RB, RBR, TOs, UH, VBN, VBP, Quotes, questionmarks, pronouns2nd, compsupadjadv, present3rd
4. Cuckoo	18	DT, JJS, LS, NNPS, PRP$, RB, RBR,TOs, VBN, VBP, VBZ, WP$, Quotes, questionmarks, exclamationmarks, pronouns2nd, imperative, present3rd

Table 6: Classification results of 1st-stage

		Proposed technique			
Classifier	Performance index	Ant	Bat	PSO	Cuckoo
MLP	Precision	0.804	0.805	0.795	**0.809**
	ROC	0.834	0.835	**0.846**	0.838
	Kappa error	0.573	**0.575**	0.558	0.584
SimpleLogistic	Precision	**0.826**	0.821	0.824	0.813
	ROC	0.861	0.862	0.862	**0.866**
	Kappa error	**0.612**	0.601	0.607	0.582
k-NN	Precision	0.740	0.752	0.733	**0.760**
	ROC	0.717	0.730	0.711	**0.738**
	Kappa error	0.438	0.465	0.423	**0.482**
RandF	Precision	**0.829**	0.819	0.824	0.826
	ROC	**0.879**	0.870	**0.879**	0.872
	Kappa error	**0.630**	0.609	0.619	0.625
C4.5	Precision	0.775	0.771	0.780	**0.787**
	ROC	0.738	0.727	0.724	**0.756**
	Kappa error	0.511	0.504	0.523	**0.540**

Table 6 demonstrates the comparative results of the first phase's classification performance to detect the most significant features. The best performance of the proposed first-stage feature reduction scheme was achieved when using RandF with the Ant algorithm (82.9%). RandF and SimpleLogistic classifiers achieved similar results with the PSO and Ant algorithms (82.4 and 82.6%, respectively). RandF achieved a precision rate of 82.9% when the Ant algorithm selected only 44% of the features, whereas *MLP*, SimpleLogistic, *k-NN* and C4.5 achieved precision rates of 80.7, 82.6, 74 and 77.5%, respectively. RandF achieved a precision rate of 81.9% when the Bat algorithm eliminated 36.5% of the features, while *MLP*, SimpleLogistic, *k-NN* and C4.5 achieved precision rates of 80.5, 82.1, 75.2 and 77.1%, respectively.

However, RandF achieved a precision rate of 82.6% when the Cuckoo algorithm selected only 34% of the features. The other classifiers, *MLP*, SimpleLogistic, *k-NN* and C4.5, achieved precision rates of 79.5, 82.4, 73.3 and 78%, respectively. When comparing classifiers, RandF outperformed the other classifiers by achieving more significant improvement, particularly when it was combined with feature reduction. However, this accuracy is lower than the best accuracy achieved on this database, which indicates the importance of feature reduction for eliminating excessive features for all classifiers. Although the results indicated comparable outcome between Ant and Cuckoo algorithms, we preferred to proceed with Cuckoo search based on Lévy flights in the next stage, as it has quick and efficient convergence, less complexity, easier to implement with a smaller number of parameters compared to PSO, Ant and Bat algorithms (Beheshti and Shamsuddin, 2013; Kamat and Karegowda, 2014).

Figure 2 shows the feature selection techniques' agreements. The Venn diagram shows that the three feature-selection approaches share 12 features according to the results generated. The 12 common elements in Ant, Bat and PSO are the frequencies of foreign words, modal auxiliaries, singular common nouns, pre-determiners, comparative adverbs, superlative adverbs, particles, base form verbs, WH-determiners, first-person pronouns, second-person pronouns and words with an objective.

Next, the best-reduced dataset feature is presented with the proposed Tabu search approach, which further optimizes the data dimensions and finds an optimal set of features. At the end of this step, a subset of features is chosen for the next round. The optimal features of the Tabu search technique are shown in Table 7.

The number of features was remarkably reduced, so less storage space is required to execute the classification algorithms. This step helped in reducing the size of the dataset to only 11 to 17 attributes. After applying the Tabu search in the second phase, the RandF classifier outperformed the other classifiers when comparing classifiers. It achieved a precision rate of 83.1%, which validates the features selected by the proposed reduction technique.

The Cuckoo search feature-selection technique enhanced the performance in most cases. Table 8 demonstrates the comparative results of the second phase's classification performance using the Tabu search algorithm to detect the most significant features. RandF classifier achieved the highest precision rate (83.1% with 11 features). The Tabu search helped in reducing the dimension of features and improved the classification performance.

Table 8 also shows the final classification results of the proposed technique for mining sports article data. The SMOTE technique was applied to the reduced dataset to increase the samples of the minority class. The training set was resized using SMOTE at an over-sampling rate of 200% to balance the number of instances in the two classes. This step contributes to making the dataset more diverse and balanced. The highest precision rate is associated with RandF and the suggested feature selection technique (89.5% with an 80% reduction in features). This method outperforms the classification results when using all the features. The results demonstrate that these features are sufficient to represent the dataset's class information.

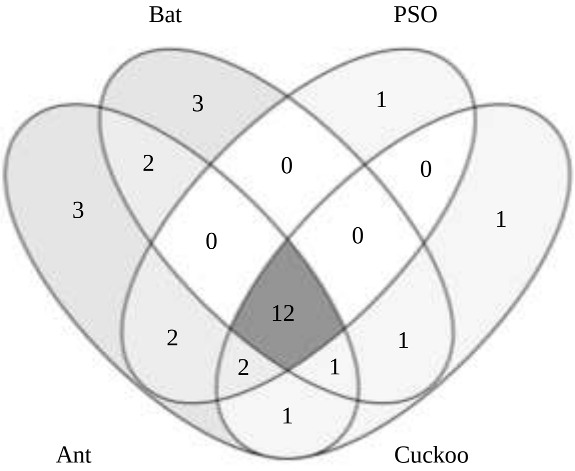

Fig. 2: Agreement of feature selection techniques

Table 7: Selected features of 2nd phase: Cuckoo then Tabu search

FS technique	Number of selected features	Selected features
TabuSearch	11	DT, JJS, LS, NNPS, PRP$, TOs, VBP, Quotes, questionmarks, pronouns2nd, imperative

Table 8: Classification results of 2nd stage before and after SMOTE

Classifier	*Performance index*	*PSO-GA*	*Cuckoo-Tabu search*
Classification results of 2nd stage before SMOTE			
MLP	*Precision*	0.823	0.823
	ROC	0.868	0.866
	Kappa error	0.617	0.618
SimpleLogistic	Precision	0.824	0.825
	ROC	0.867	0.866
	Kappa error	0.603	0.606
k-NN	Precision	0.766	0.766
	ROC	0.747	0.747
	Kappa error	0.4953	0.495
RandF	Precision	0.820	0.831
	ROC	0.871	0.871
	Kappa error	0.611	0.635
C4.5	*Precision*	0.774	0.774
	ROC	0.760	0.763
	Kappa error	0.511	0.511
Classification results of 2nd-stage after SMOTE			
MLP	*Precision*		0.834
	ROC		0.873
	Kappa error		0.641
SimpleLogistic	*Precision*		0.839
	ROC		0.877
	Kappa error		0.650
k-NN	*Precision*		0.846
	ROC		0.823
	Kappa error		0.661
RandF	*Precision*		0.895
	ROC		0.948
	Kappa error		0.774
C4.5	*Precision*		0.856
	ROC		0.856
	Kappa error		0.689

Discussion

Cuckoo and Tabu search helped in improving the classification performance with a limited number of features. In terms of precision, ROC and Cohen's kappa coefficient, the proposed technique with SMOTE significantly improved the classification accuracy of the minority class while keeping the classification accuracy of the majority class high. The nine common features according to the results generated using Cuckoo-Tabu search and the three techniques were the frequencies of foreign words, modal auxiliaries, singular common nouns, pre-determiners, comparative adverbs, base form verbs, WH determiners, first-person pronouns and second-person pronouns (Fig. 3).

Table 9 shows the effect of classification using the nine common features between the At, Bat, PSO and Cuckoo-Tabu search techniques. The scored results are not better than those in the first phase. The results from the suggested two-stage attribute selection phase show better performance than those of datasets that were not preprocessed and when these attribute selection techniques are used independently. Moreover, the results are better than those on the same dataset with an approach using a modified Cortical Algorithm (CA) (Sait and Youssef, 1999). That approach had an accuracy of 85.6% with a 40% reduction in features.

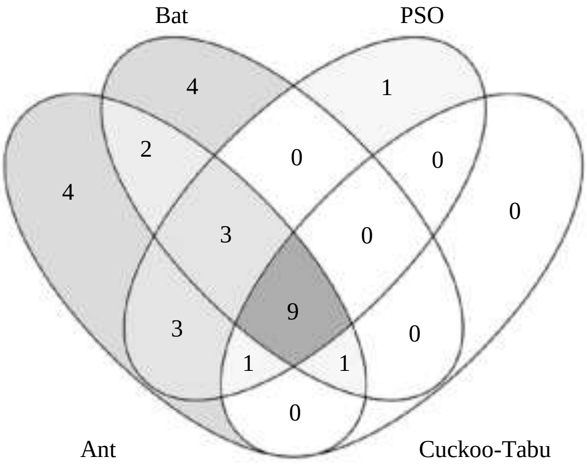

Fig. 3: Agreement of feature selection techniques

Undoubtedly, the SMOTE resampling technique can effectively enhance the classifier performance, as any classification model can obtain higher accuracy if applied to a balanced dataset. Under most imbalanced circumstances, the re-balancing methods were worthwhile.

Table 9: Classification results *Ant, Bat, PSO and Cukoo-Tabu search*

Classifier	Performance index	9 common features
MLP	Precision	0.815
	ROC	0.861
	Kappa error	0.600
SimpleLogistic	Precision	0.819
	ROC	0.865
	Kappa error	0.595
k-NN	Precision	0.758
	ROC	0.737
	Kappa error	0.478
RandF	Precision	0.828
	ROC	0.866
	Kappa error	0.627
C4.5	Precision	0.789
	ROC	0.777
	Kappa error	0.5435

Conclusion

In this study, we are motivated to study the impact of suggesting a two-stage heuristic feature selection method using Tabu search and Cuckoo search with Lévy flight in proposed classifying sports articles. The experiments showed that applying Tabu search and Cuckoo search techniques helped in remarkably reducing feature numbers. The suggested model enhanced the precision performance and achieved promising results. Furthermore, altering the original data with SMOTE technique helped to increase the region of the minority class, which eventually helped with handling imbalanced data.

Acknowledgment

This work is funded by PAAET – Kuwait under Grant No:BS-19-20. The author extends their sincere thanks to the editors of this journal and the anonymous reviewers for their valuable comments and suggestions that have significantly improved the manuscript.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Abbasi, A., France, S., Zhang, Z., & Chen, H. (2011). Selecting attributes for sentiment classification using feature relation networks. In IEEE Transactions on Knowledge and Data Engineering, 23, 447-462. https://doi.org/10.1109/TKDE.2010.110

Beheshti, Z., & Shamsuddin, S. M. H. (2013). A review of population-based meta-heuristic algorithms. Int. J. Adv. Soft Comput. Appl, 5(1), 1-35. https://www.researchgate.net/profile/Zahra-Beheshti-4/publication/270750820_A_review_of_population-based_meta-heuristic_algorithm/links/54b3abd60cf28ebe92e2fa2e/A-review-of-population-based-meta-heuristic-algorithm.pdf

Ben-David, A. (2008). Comparison of classification accuracy using Cohen’s Weighted Kappa. Expert Systems with Applications, 34(2), 825-832. https://doi.org/10.1016/j.eswa.2006.10.022

Breiman, L. (1996). Bagging predictors. Machine learning, 24(2), 123-140. Breiman, L. (2000). Some infinity theory for predictor ensembles. Technical Report 579, Statistics Dept. UCB. https://doi.org/10.1007/BF00058655

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32. Breiman, L. (2004). Consistency for a simple model of random forests. Statistical Department. University of California at Berkeley. Technical Report, (670).

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, 321-357. https://doi.org/10.1613/jair.953

Chen, W., Chen, H., Feng, Q., Mo, L., & Hong, S. (2021). A hybrid optimization method for sample partitioning in near-infrared analysis. Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy, 248, 119182. https://doi.org/10.1016/j.saa.2020.119182

Civicioglu, P., & Besdok, E. (2013). A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. Artificial intelligence review, 39(4), 315-346. https://doi.org/10.1007/s10462-011-9276-0

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization. Machine learning, 40(2), 139-157. https://doi.org/10.1023/A:1007607513941

Glover, F. (1989). Tabu search-part I. ORSA Journal on computing, 1(3), 190-206. https://doi.org/10.1287/ijoc.1.3.190

Glover, F. (1990). Tabu search-part II. ORSA Journal on computing, 2(1), 4-32. https://doi.org/10.1287/ijoc.2.1.4

Hajj, N., Rizk, Y., & Awad, M. (2019). A subjectivity classification framework for sports articles using improved cortical algorithms. Neural Computing and Applications, 31(11), 8069-8085. https://doi.org/10.1007/s00521-018-3549-3

Heerschop, B., Hogenboom, A., & Frasincar, F. (2011). Sentiment Lexicon Creation from Lexical Resources. In 14th Int. Conf. on Business Information Systems, 87, 185–196. https://doi.org/10.1007/978-3-642-21863-7_16

Kamat, S., & Karegowda, A. G. (2014). A brief survey on cuckoo search applications. Int. J. Innovative Res. Comput. Commun. Eng, 2(2). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1070.1007&rep=rep1&type=pdf

Kaveh, A., & Bakhshpoori, T. (2013). Optimum design of steel frames using Cuckoo Search algorithm with Lévy flights. The Structural Design of Tall and Special Buildings, 22(13), 1023-1036. https://doi.org/10.1002/tal.754

Korycinski, D., Crawford, M., Barnes, J. W., & Ghosh, J. (2003, July). Adaptive feature selection for hyperspectral data analysis using a binary hierarchical classifier and tabu search. In IGARSS 2003. 2003 IEEE International Geoscience and Remote Sensing Symposium. Proceedings (IEEE Cat. No. 03CH37477) (Vol. 1, pp. 297-299). IEEE. https://doi.org/10.1109/IGARSS.2003.1293755

Liu, B. (2010). Sentiment analysis and subjectivity. Handbook of natural language processing, 2, 627- 666.

Liu, B. (2012). Sentiment Analysis and Opinion Mining, Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers. https://doi.org/10.2200/S00416ED1V01Y201204HLT016

Pang, B. & Lee, L. (2008). Opinion Mining and Sentiment Analysis, Foundations and Trends in Information Retrieval 2, 1–135. https://doi.org/10.1561/1500000011

Pang, B., Lee L. & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing-Volume 10. Association for Computational Linguistics. https://doi.org/10.3115/1118693.1118704

Rizk, Y., & Awad, M. (2012, August). Syntactic Genetic Algorithm for a Subjectivity Analysis of Sports Articles. In 11th IEEE International Conference on Cybernetic Intelligent Systems, Limerick, Ireland.

Rodrigues, D., Pereira, L. A., Almeida, T. N. S., Papa, J. P., Souza, A. N., Ramos, C. C., & Yang, X. S. (2013, May). BCS: A binary cuckoo search algorithm for feature selection. In 2013 IEEE International Symposium on Circuits and Systems (ISCAS) (pp. 465-468). IEEE. https://doi.org/10.1109/ISCAS.2013.6571881

Sait, S. M., & Youssef, H. (1999). General iterative algorithms for combinatorial optimization. IEEE Computer Society.

Schumacher, M., Holländer, N., & Sauerbrei, W. (1997). Resampling and cross-validation techniques: a tool to reduce bias caused by model building?. Statistics in medicine, 16(24), 2813-2827. https://doi.org/10.1002/(SICI)1097-0258(19971230)16:24<2813::AID-SIM701>3.0.CO;2-Z

Smialowski, P., Doose, G., Torkler, P., Kaufmann, S., & Frishman, D. (2012). PROSO II–a new method for protein solubility prediction. The FEBS journal, 279(12), 2192-2200. https://doi.org/10.1111/j.1742-4658.2012.08603.x

Srivastava, P. R., Khandelwal, R., Khandelwal, S., Kumar, S., & Ranganatha, S. S. (2012). Automated test data generation using cuckoo search and tabu search (CSTS) algorithm. https://doi.org/10.1515/jisys-2012-0009

Tahir, M. A., Bouridane, A., Kurugollu, F., & Amira, A. (2004a, August). Feature selection using tabu search for improving the classification rate prostate needle biopsies. In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. (Vol. 2, pp. 335-338). IEEE. https://doi.org/10.1109/ICPR.2004.1334201

Tahir, M.A., Bouridane, A. & Kurugollu, F. (2004b). Simultaneous Feature Selection and Weighting for Nearest Neighbor Using Tabu Search. In: 5th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL 2004), Exeter, UK. https://doi.org/10.1007/978-3-540-28651-6_57

Turney, P. D. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics. https://doi.org/10.3115/1073083.1073153

Wiebe, J., & Riloff, E. (2011). Finding mutual benefit between subjectivity analysis and information extraction. IEEE Transactions on Affective Computing, 2(4), 175-191. https://doi.org/10.1109/T-AFFC.2011.19

Wiebe, J., Wilson, T., Bruce, R., Bell, M., & Martin, M. (2002). Learning subjective language. In Technical Report TR-02-100, Department of Computer Science, University of Pittsburgh, Pittsburgh, Pennsylvania.

Yang, X. S., & Deb, S. (2009, December). Cuckoo search via Lévy flights. In 2009 World congress on nature & biologically inspired computing (NaBIC) (pp. 210-214). IEEE. https://doi.org/10.1109/NABIC.2009.5393690

Yang, X. S., & Deb, S. (2014). Cuckoo search: recent advances and applications. Neural Computing and Applications, 24(1), 169-174. https://doi.org/10.1007/s00521-013-1367-1

Yu, H., & Hatzivassiloglou, V. (2003). Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In Proceedings of the 2003 conference on Empirical methods in natural language processing (pp. 129-136). https://doi.org/10.3115/1119355.1119372

Zhou, X., Tao, X., Yong, J., & Yang, Z. (2013). Sentiment analysis on tweets for social events. In Computer Supported Cooperative Work in Design (CSCWD), 2013 IEEE 17th International Conference on (pp. 557-562). IEEE. https://doi.org/10.1109/CSCWD.2013.6581022