Journal of Engineering Science and Technology
Vol. 14, No. 3 (2019) 1601 - 1613

AN OPTIMISED SUPPORT VECTOR MACHINE WITH RINGED SEAL SEARCH ALGORITHM FOR EFFICIENT TEXT CLASSIFICATION

WAREESA SHARIF[1], IWAN TRI YADI YANTO[2],
NOOR AZAH SAMSUDIN[1], MUSTAFA MAT DERIS[1], ABDULLAH KHAN[3],
MUHAMMAD FAHEEM MUSHTAQ[4], MUHAMMAD ASHRAF[5]

[1]Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn, Malaysia (UTHM), Johor, Malaysia
[2]Information System Department,
University of Ahmad Dahlan, Jalan Kapas n 9, Yogyakarta, 55165, Indonesia
[3]Department of Computer Science and Information Technology,
Agriculture University Peshawar, Pakistan
[4]Department of Information Security, Khwaja Fareed University of Engineering
and Information Technology, Rahim Yar Khan, 64200, Pakistan
[5]Department of Management Sciences,
COMSATS University Islamabad, Pakistan.
*Corresponding Author: wareesa786@gmail.com

Abstract

Nowadays, with the increasing availability of online text documents, it becomes an important task for an organization to automatically classify the document. In Text Classification (TC), Support Vector Machine is the commonly used machine-learning algorithm. Performance of SVM highly depends on parameter tuning using metaheuristic algorithm for text classification. To integrate dynamic searching to parameter setting for SVM is a big issue that produced great influence in the classification accuracy. In order to improve the generalization and learning capability of SVM, this paper presents a new approach known as RSS-SVM, which is used to optimize kernel function and penalty parameters through the Ringed Seal Search algorithm. Experiments are conducted on three text datasets named: Reuter21578, 20 newsgroup and TDT2 with a different number of classes, which shows that proposed RSS-SVM present significant results having 79.22% accuracy, 70.79% recall, 58% precision and 54.71% f-measure among the previous GA-SVM and CS-SVM algorithms.

Keywords: Metaheuristic, Parameter optimisation, Ringed seal search, Support vector machine, Text classification.

1. Introduction

Text data are increasing rapidly. Consequently, classification is a challenging task in typical text documents because of the ever-increasing amount of electronic documents, web resources and digital libraries [1]. A lot of struggle has been done in this area, however, it remains an open issue. That is why, text classification becomes important to label documents into predefined classes [2, 3]. Text classification is one of these fields in which, researchers are working to improve classification accuracy because text data are very high dimensional data. Classifying high dimensional data and improve its accuracy is an important scientific problem. Ageev et al. [4] explained that in these days, the best text classification systems use the machine learning approach, which produced better precision, recall and *f*-measure. The support vector machine is a better technique compared to other machine learning techniques. SVM is used to apply in various problem domain such as bioinformatics [5, 6], sentiment analysis [7], online handwritten recognition [8], text classification [4, 9-11]. A big challenge in adopting SVM for practical real-world problems relies on parameter selection. Multiple SVMs with different parameters (*C*, gamma) have to be computed in order to produce better classification performance. Therefore, various studies have been shown that the kernel parameter and penalty factor affect the accuracy of SVM in the classification task very seriously. When the values of the penalty factor and kernel function parameter are selected appropriately, the classification accuracy of SVM can be improved.

Based on studies by Gaspar et al. [12], optimisation can be used to get the general power of kernel function. Many kernel functions are used to tune the SVM parameter for text classification in which, Gaussian RBF kernel is best to improve the performance of optimisation techniques. Many metaheuristic techniques, which plays an important role in various domains optimisation problem. These metaheuristic techniques copy natural phenomena from millions of year [5, 13]. The advantage of metaheuristic techniques is that it maintains good performance with dynamic changes. The most commonly known metaheuristic techniques used in data mining are Genetic algorithm (GA) [5], Cuckoo Search (CS) [14], Particle Swarm Optimisation (PSO) [13]. GA is perceived as one of the important approaches using operators enthused by natural selection and genetic variation [5].

According to Eberhart and Kennedy [15], another optimisation technique is PSO is proposed by Aghdam and Heidar [16], in which, stimulated by the fish and bird swarm intelligence. Further, brood intelligent behaviour of some cuckoo species inspires Cuckoo Swarm [17]. A cuckoo lays its eggs in other cuckoos' nests. These search algorithms have been used to optimize the parameter of a support vector machine to improve performance.

This paper examines the effect of different parameters of SVM on a text classification performance. GA-SVM can automatically select parameters to produce an optimal gene subset. Due to premature convergence for high dimensional complex problems, original Cuckoo Swarm (CS) optimisation falls into local optimum. CS, PSO, and GA are dominating global optimisation algorithms used in science and technology applications. To find new solutions, they have some limitations to maintain the balance between exploration and exploitation [17]. CS-SVM cannot provide a strong mechanism for optimal balance between exploration and exploitation that is why it cannot set the parameter strongly. Ringed Seal Search algorithm is metaheuristic with two search states (Brownian & Levy) that alternate randomly due to noise and provide a balance between exploitation and exploration of the search, and therefore, the probability to get local optima easily is very low. Furthermore, the parameter used in RSS is comparatively less than GA, PSO and CS.

This research describes an algorithm to find a strong mechanism to optimal parameters, which balance exploitation and exploration. RSS is able to select γ and C parameters of SVM properly. RSS-SVM showed a significant impact of optimizes parameter for SVM with RSS and provide better accuracy in text classification. The proposed algorithm estimates the bounds for searching optimal parameters. The range for searching the parameters depends on a number of positive examples. They used Reuters-21578, 20 newsgroup and TDT2 document collection that is specially developed for text categorisation researches. The proposed model shows substantial performance improvement on a more complex task, such as text categorisation for large systems of categories. The rest of the article is organized as follows. In Section 2, the Ringed Seal Search algorithm is presented. In Section 3, the proposed RSS-SVM is illustrated. Furthermore, the performance of the classifier is discussed in detail. Section 4 describes the experimental setup in the conclusion. Result and discussion are illustrated in Section 5 and conclusion is described in Section 6.

2. Ringed Seal Search (RSS) Algorithm

RSS is a metaheuristic technique that is proposed to solve global optimisation problems. RSS technique is based on the seal pups' behaviour for finding the best lair to escape predators. This model divides the search into two states such as normal state and urgent state. Pups do intensive and extensive search under normal state and urgent state respectively [18]. Every time, a seal pup finds good quality lair and move in it.

If the current state of the search space ρ is ω where $\omega = 1$ (ω represents the external noise), then ∂ is informed that Ω contains β, which is a predator emitting a noise ω during movement. Given E event in Ω, a state (Ω, ρ) is called urgent state, if Ω includes β and ∂ members of the event at the search space that contains the noise ω. Let A be an event where $(\Omega, \beta, \partial, \rho)$ is the search space. If the current state of the search space ρ is ω where $\omega = 0$, then ∂ is not informed that Ω, contains β, then (Ω, ρ) is considered as a normal state. For urgent state ∂ performs a Levy walk and for normal state ∂ performs a Brownian walk. The next section will discuss the proposed RSS-SVM.

3. Proposed RSS-SVM Algorithm

This paper proposed a novel Ringed Seal Search based Support Vector Machine (RSS-SVM) for the text classification. The RSS is a metaheuristic technique that is used to optimise the kernel function and penalty parameter of the support vector machine to improve the performance of existing GA-SVM and CS-SVM algorithms. In this paper, One Versus All (OVA) approach is used. One of the main challenges to optimised SVMs is the parameter selection for the classification task. It is a common procedure to get better classification results by optimizing SVMs using nature-inspired metaheuristic search algorithms. Following is the brief description for the SVM classifier. The training set of SVM has inputs such as $T = \{(x_{i,}y_i).....(x_{i,}y_i)$ where $x_i \in R^2$ and $y_i \in \{-1,1\}$.

The main objective is to optimise the SVM and to find a hyperplane that accurately classifies the training dataset into two categories. The SVM classification problem is constructed as follows:

$$\min_{w,b,\xi} \frac{1}{2}\|x_i\|^2 + c\sum_{i=1}^{1}\xi_i$$

$$y_i((w.\Phi(x_i)) + b) \geq 1 - \xi_i \tag{1}$$

$$\xi_i \geq 0, i = 1.....................1$$

where $C > 0$ is the penalty parameter of the error term $\xi i\phi(.)$. In Eq. (1), w and b are the normal vector and the offset of the separating hyperplane, respectively. Following is the translation of Eq. (2) to the Lagrange dual problem.

$$\min_a \frac{1}{2}\sum_{i=1}^{1}\sum_{j=1}^{1} y_i\, y_j\, a_i\, a_j \left(\varphi(x_i).(x_j)\right) - \sum_{j=1}^{1} a_j \quad \text{s.t}$$

$$\sum_{i=1}^{l} y_i a_j = 0 \tag{2}$$

$$0 \leq a_i \leq$$

$$C\, i = 1.........1$$

From Eq. (2), the Lagrange multipliers $\alpha i \in (0, C)$ are obtained, and the classification decision function (x) is then constructed as in Eq. (3).

$$f(x) = sign\left(\sum_{i=1}^{1} a_i y_i \left(\varphi(x_\text{i}).\varphi(x_\text{j})\right) + b\right) \tag{3}$$

Generally, the kernel function is defined as $K(x_i, x) = (\varphi(x_i).\, \varphi(x))$. There are many kernel functions used by SVM for the classification task. The most used kernel function is the Gaussian RBF kernel function. With RBF, classification of SVM performance is better as compared with the other kernel functions. Therefore, this paper selects the Gaussian RBF kernel function is as follow:

$$k(x_i, y_i) = \exp\left(-\gamma\left\|x_i - x_j\right\|^2\right) \tag{4}$$

where within Gaussian RBF kernel function in Eq. (4), γ is the kernel parameter. Normally, the kernel parameter γ and the penalty parameter C is referring to the SVM parameters with the Gaussian RBF kernel function, which should be optimised by the user. Ringed Seal Search (RSS) is used to optimise of SVM parameters. The RSS is based on the search behaviour of seal pups to find best lair to escape predators. The sensitive search model inspired by seal movement is introduced by the proposed RSS algorithm. Every time seal pup moves in good quality lairs. These lairs provide protection from predators (e.g., bears) and also thermal protection against cold air temperatures and high wind chill. A seal could have a complex of lairs at a specific area. A series of events can be described during the search for new lair or the movement of the seal pup inside its multi-chambered lair. By modifying a random value, the evolution is achieved. To find the best combinations of parameters in each iteration, the selected parameters are placed in a vector form and that vector is evolved the initial population for SVM parameters is represented by a matrix.

Nature inspired RSS always starts with initial values to solving an optimisation problem. Optimisation processes consist of a vector of values ($L_i, i=1,2,3\ldots n.$) that represent the initial solution. The RSS algorithm always starts with an initial number of birthing lairs *n*, which consists of multi-chambered. Pups move into a search space for finding a new lair with better quality. To find better search space it is necessary to form an array from these initial values in the search space. The number of lairs in the RSS algorithm that represents the lairs for seal pup is defined as in Eqs. (5) and (6):

$$L_i, i=1,2,3,\ldots n \tag{5}$$

Lairs are randomly distributed, and each lair *i* contains chambers *m*. For instance, an array of $L=[i \times m]$ representing current existing lair *i* of a habitat for a lair *i*.

$$L=\left[i \times m\right] \tag{6}$$

The values are distributed between predefined lower bound L_{bj} and upper bound U_{bj} randomly and uniformly at the search space that is described in Eq. (7).

$$L_i = L_b + \left(U_b + \right).rand\left(size\left(L_b\right)\right) \tag{7}$$

$$where\ i=1,2,3,\ldots,n$$

where *n* indicates the number of the initialized lairs and *i* represents the number of the lair. In a specific search pattern, the seal moves from a lair to a new lair, which generates new solutions (new lairs) x^{t+1} for seal *i*, a new lair is found in Eq. (8):

$$x_i^{t+1} = x_i^t + a * \Delta x \tag{8}$$

where α is the step size during normal or urgent states.

$$\Delta x = {}^{\lambda}levy \quad \text{where w = 1} \tag{9}$$

where ω represents a uniform discrete distribution shown in Eq. (9). In case of Levy walk, the random walk is characterized by a step size calculated from a probability distribution with an inverse power-law tail as shown in Eq. (10).

$$Levy \sim u = \overline{t^{-\lambda}} \tag{10}$$

where $1< \lambda < 3$ and *t* is the flight length. In the case where the value of $\lambda \geq 3$, the distribution will not be in a heavy tail and the total sums of the lengths converge to a Gaussian distribution.

Levy walk is described by an anomalous diffusion in which, the mean squared displacement increases linearly fast with time. Contrary to levy walk, Brownian walk is characterised with a normal diffusion in which, the mean squared displacement increases linearly.

Structure of Brownian walk search for a new chamber inside the multi-chambered lair structure, as shown in Eq. (11).

$$\Delta x = {}^{\lambda} browni \text{ where } w = 0 \tag{11}$$

The search is characterised by a step size described as in Eq. (12).

$$S = K * rand(d, Ndot) \tag{12}$$

where *K* is the standard deviation of the normal distribution is for diffusion rate coefficient is *d*, the dimensions of the problem and *Ndots* represents the number of particles of the Brownian in the search space.

4. Experimental Setup

A series of experiments have been conducted to check the efficiency of RSS-SVM algorithm with three datasets obtained from UC Irvin machine learning repository: Reuter-21578 [11] and 20 newsgroup [19] and TDT2. For Reuter-21578 dataset, 15 classes are selected. From 20 newsgroup dataset, 10 classes and for TDT2, 5 classes are selected. All datasets are single label datasets. Feature selection is used to select the best features for classification. In this paper, the Improved Relative Discriminative Criterion (IRDC) feature ranking technique is used [7]. Moreover, not all features play a positive role, and some might even contribute negatively to the classification process. It becomes important to select the best subset of features that improves the ability of the SVM to generalize the model. Thus, many SVM optimisation strategies focus on the process of Feature Selection (FS) [11, 20, 21]. Program for proposed RSS-SVM and existing algorithms are written in MATLAB software. These datasets are chosen on the basis of their popularity. Datasets are divided into two sections. 70% data is used for training and 30% is chosen for testing. The pseudo-code of the proposed RSS-SVM algorithm is presented in Fig. 1.

```
Begin
Initialised SVM parameter and structure
Generate an initial number of birthing lairs
L_1 = (f = 1,2,3.......n)
While ( Stopping criterion)
If noise = false
Search in the proximity for a new lair by using a Brownian walk
Else
Expend the search for a way for a new layer by using levy walk
End if
Evaluate the fitness of each new lair and compare with previous
If
L^{best,t} > L^{best,k+1}
Choose the new lair
L^{best} = L^{best,k}
Else
Go to 4
End if
Rank the solutions;
Return the best lair
The global best lair is fed to SVM classifier for training
Training the SVM classifier
End while
End
```

Fig. 1. Pseudo-code of proposed RSS-SVM algorithm.

Performance criteria

Tang and Liu [22, 23] reported that in text classification, accuracy is not the only measuring criteria to evaluate the performance of a classification model, whereas precision, recall, and *f*-measure are also important. Precision is computed as in Eq. (13) and Recall is presented in Eq. (14). Accuracy and *F*-measure are shown in Eqs. (15) and (16).

$$precision = \frac{tp}{tp + fp} \tag{13}$$

whereas *tp* denote the true positive rate and *fp* show the false positive rate in precision.

$$Recall = \frac{tp}{tp + fn} \tag{14}$$

whereas, *tp* describe the true positive rate and *fn* denote the false negative rate in the recall.

Accuracy is the ratio between the numbers of correctly classified objects over the total number of objects. Inaccuracy, true positive (*tp*), true negative (*tn*), false negative *(fn)* and false positive (*fp*) values are calculated as in Eq. (15):

$$Accuracy = \frac{tp + tn}{tp + fp + tn + fn} \tag{15}$$

F-measure is the harmonic mean in which, precision and recall are combined, and the traditional f-measure is calculated as in Eq. (16):

$$F - measure = 2 * \frac{p.r}{p + r} \tag{16}$$

where, *p* denotes the precision and r show the recall in F-measure.

5. Result and Discussion

To analysis the result, different experiments are conducted and the performance of RSS-SVM is compared with CS-SVM and GA-SVM. Different measuring criteria such as Accuracy, F-measure, Precision, and Recall are used to check the performance on three datasets: such as reuter21578, 20 newsgroup and TDT2. A different number of classes is used for testing the proposed model. The algorithm stops if there is no improvement in the objective function for 2000 seconds or the improvement is less than tol = 5*e*-4.

5.1. Result of reuter-21578 dataset

These series of experiments are tested on different classes as 2-5, 8, 10, 12, and 15 that can be seen in Table 1. These experiments show that the performance of the proposed technique is better than previous techniques. For Reuter-21578, the accuracy of RSS-SVM is better than GA-SVM and CS-SVM on 2-5, 8, 10, 12, and 15 classes. On the whole dataset, RSS-SVM produces significant result than previous techniques. RSS-SVM produced 25.98% accuracy while GA-SVM generated 19.95% and CS-SVM presented 18.79% on text dataset. RSS-SVM shows better *F*-measure than GA-SVM and RS-SVM. In case of precision, RSS-SVM produced a better result for all classes except four and five classes. Precision for 10 classes only, RSS-SVM and GA-SVM produced an equal result. In the case of a recall, RSS-SVM produced significant result compare to existing GA-SVM and CS-SVM techniques. On the whole dataset, accuracy is good than previous techniques that are shown in Table 1 and Fig.2. Precision, recall and *f*-measure also checked that shown better performance for RSS-SVM technique as compared to GA-SVM and CS-SVM algorithms.

Table 1. Performance of the RSS-SVM among GA-SVM and CS-SVM for Reuters-21578 dataset.

Classifiers	Measure criteria	No. of classes 2	3	4	5	8	10	12	15
GA-SVM	Accuracy	0.58	0.4833	0.6923	0.6923	0.4706	0.4548	0.4789	0.1995
	F-measure	0.49	0.3983	0.6389	0.6389	0.0714	0.0429	0.0514	0.0267
	Precision	0.58	0.58	0.6541	0.6541	0.16	0.1093	0.1244	0.1055
	Recall	0.7717	0.6761	0.634	0.634	0.391	0.391	0.1226	0.0821
CS-SVM	Accuracy	0.1879	0.1879	0.1879	0.1879	0.2871	0.2579	0.4789	0.1879
	F-measure	0.0367	0.0397	0.0397	0.0367	0.0367	0.0327	0.0514	0.0267
	Precision	0.1055	0.1055	0.1097	0.2055	0.1095	0.1055	0.1204	0.1055
	Recall	0.0917	0.0917	0.271	0.271	0.371	0.371	0.1236	0.0817
RSS-SVM	Accuracy	0.6701	0.511	0.8077	0.8077	0.4706	0.4561	0.4789	0.2598
	F-measure	0.6297	0.4298	0.6641	0.6641	0.0714	0.0429	0.0514	0.0587
	Precision	0.6701	0.6	0.6429	0.6429	0.26	0.1093	0.1294	0.1197
	Recall	0.8012	0.6786	0.8958	0.8958	0.5609	0.5083	0.1549	0.0999

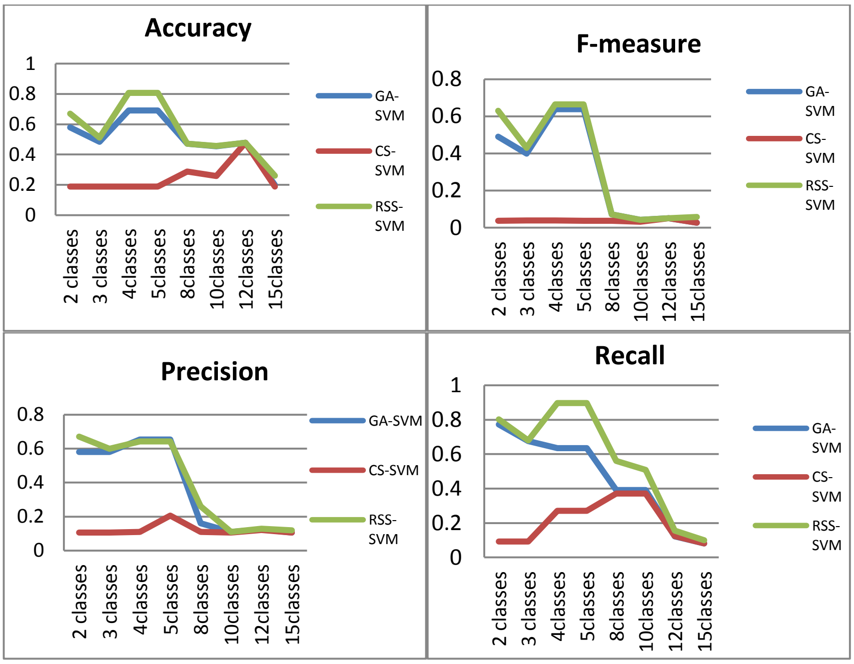


Fig. 2. Convergence performance of the RSS-SVM among GA-SVM and CS-SVM algorithms for Reuter-21578.

5.2. Result of 20 newsgroup dataset

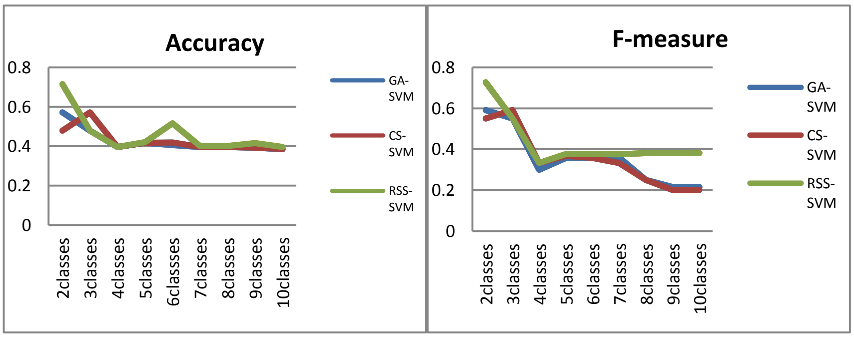
The experiment is conducted on 20 newsgroup text dataset with a different number of classes, which shows that the proposed RSS-SVM technique provides a better result than existing GA-SVM and CS-SVM techniques. For 2-10 classes, RSS-SVM technique provides better accuracy, however, only for three or four classes, it provides an equal result with GA-SVM. RSS-SVM produced 39.64%.

Accuracy while GA-SVM showed 38.49% and CS-SVM produced 38.59%. RSS-SVM also produced better *F*-measure for all classes that are 12.35% while GA-SVM presented 08.87% and 08.01%. The result on 20 newsgroup is also evaluated for precision, all classes for RSS-SVM technique produced better result compare existing techniques except three classes.

The precision of RSS-SVM produced 38.15% while GA-SVM showed 20.10% and CS-SVM presented 20.10%. For recall, the result for RSS-SVM is better than that of GA-SVM and CS-SVM techniques presented in Table 2 and Fig. 3.

Table 2. Performances RSS-SVM result among GA-SVM and CS-SVM of the 20 newsgroup dataset.

Classifiers	Measure criteria	No. of classes								
		2	3	4	5	6	7	8	9	10
GA-SVM	Accuracy	0.5714	0.4793	0.3957	0.4176	0.406	0.3967	0.3967	0.393	0.3849
	F-measure	0.4987	0.3592	0.1361	0.1755	0.1601	0.1472	0.0984	0.0928	0.0887
	Precision	0.5909	0.55	0.3002	0.3576	0.3588	0.3588	0.25	0.215	0.215
	Recall	0.7632	0.6833	0.0855	0.4028	0.3918	0.3702	0.31	0.2661	0.2799
CS-SVM	Accuracy	0.4793	0.5714	0.3957	0.4176	0.419	0.3967	0.3967	0.393	0.3859
	F-measure	0.3592	0.4987	0.1352	0.1755	0.1655	0.1401	0.0984	0.0928	0.0801
	Precision	0.55	0.5909	0.3333	0.3667	0.3575	0.3333	0.25	0.201	0.201
	Recall	0.6833	0.7632	0.0855	0.4028	0.3918	0.3702	0.31	0.2661	0.2799
RSS-SVM	Accuracy	0.7143	0.4793	0.3957	0.4201	0.517	0.4014	0.4014	0.4149	0.3964
	F-measure	0.6971	0.3592	0.1361	0.1755	0.1691	0.1522	0.1381	0.1301	0.1235
	Precision	0.7273	0.55	0.3333	0.3767	0.3767	0.3751	0.3806	0.3815	0.3815
	Recall	0.8125	0.6833	0.0855	0.4028	0.3918	0.3831	0.3772	0.3721	0.3681



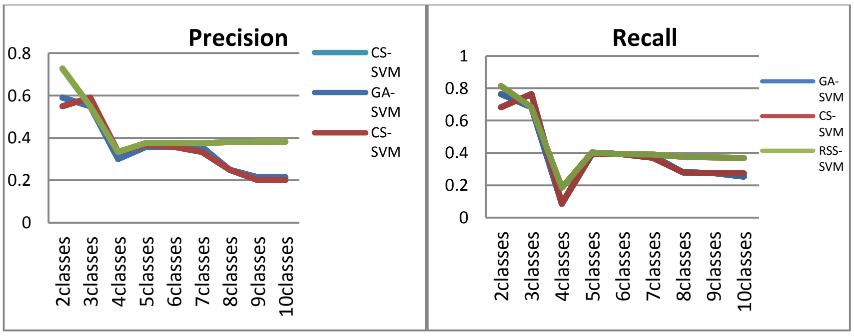


Fig. 3. Convergence performance of RSS-SVM algorithm against GA-SVM and CS-SVM on 20 newsgroup dataset.

5.3. Result of TDT2 dataset

Experiments are conducted on RSS-SVM optimisation technique and performance is checked against GA-SVM and CS-SVM techniques. These experiments are conducted on a different number of classes such as 2-5. RSS-SVM is evaluated on different measures in which, accuracy, *F*-measure, Precision and Recall are included that can be seen in Table 3 and Fig.4. For 2-5 classes, RSS-SVM technique is checked, which produced better accuracy against existing GA-SVM and CS-SVM techniques. RSS-SVM produced 81.22% accuracy while GA-SVM produced 78.94% and CS-SVM presented 53.68% accuracy.

In the case of *F*-measure, RSS-SVM showed the significant result as 55.71% and GA-SVM produced 54.38% and CS-SVM showed 31.08%. RSS-SVM also provided better precision for all 2,3,4,5 classes. For all classes, RSS-SVM showed 59.21% precision while GA-SVM produced 57.6% and 39% precision for CS-SVM. Result of RSS-SVM is also measured for recall that presented 72.97%% result while GA-SVM showed 70.60% and CS-SVM produced 33.20%. The overall result of RSS-SVM produced a better performance against GA-SVM and CS-SVM techniques.

Table 3. Performance of RSS-SVM among GA-SVM and CS-SVM on TDT2 dataset.

Classifiers	Measure criteria	No. of classes			
		2	3	4	5
GA-SVM	Accuracy	0.6378	0.5984	0.6794	0.7894
	F-measure	0.5396	0.4015	0.4498	0.5438
	Precision	0.5400	0.4800	0.4900	0.5760
	Recall	0.5401	0.4489	0.5118	0.7060
CS-SVM	Accuracy	0.6379	0.7282	0.6837	0.5368
	F-measure	0.5371	0.3894	0.4538	0.3108
	Precision	0.5400	0.4667	0.4950	0.3900
	Recall	0.5438	0.4479	0.5192	0.3320
RSS-SVM	Accuracy	0.6808	0.6038	0.6893	0.8122
	F-measure	0.5857	0.4053	0.4582	0.5571
	Precision	0.5867	0.4867	0.5000	0.5921
	Recall	0.5969	0.4517	0.5406	0.7297

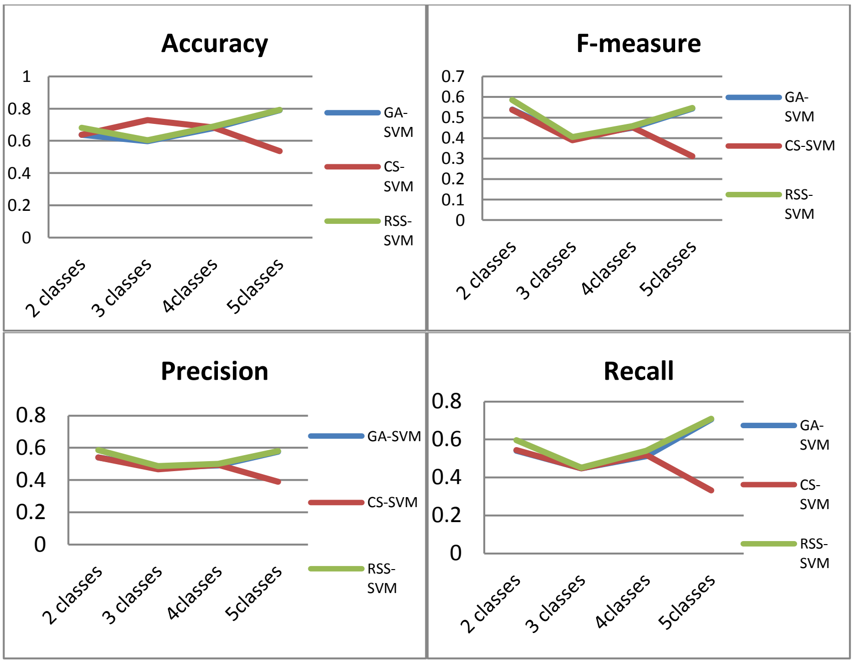


Fig. 4. Performance of the RSS-SVM among GA-SVM and CS-SVM on TDT2 dataset.

6. Conclusion

SVM is used to apply for various classification problems in a different domain, bioinformatics, sentiment analysis, online handwritten recognition and text classification. Text classification is one of these fields in which, researchers are working to improve classification accuracy. From the literature study, it showed that search algorithms affect the performance of SVM for optimisation problem in text classification. Therefore, this research proposed a new algorithm RSS-SVM that is used to optimise the parameters of SVM for better text classification accuracy. Three different datasets named as Reuter21578, 20 newsgroup and TDT2 are used to check the performance of the proposed model. From the simulation result, it showed that proposed RSS-SVM outperforms than existing algorithms in term of Accuracy, *F*-measure, Precision, and Recall. From experimental result for different classes of these three datasets, it showed that the proposed model has significant performance in term accuracy, *F*-measure, precision and recall.

Nomenclatures

C	Penalty parameter

Greek Symbols

β	Predator
∂	Seal pup
ρ, Ω	Search space

Abbreviations

IRDC	Improved Relative Discriminative Criterion
NRDC	Normalized Relative Discriminative Criterion
RDC	Relative Discriminative Criterion
RSS	Ringed Seal Search Algorithm

References

1. Paul, A. (2014). *Effect of imbalanced data on document classification algorithms*. Master Thesis. Auckland University of Technology, New Zealand.
2. Parlak, B.; and Uysal, A.K. (2016). The impact of feature selection on medical document classification. *Proceedings of the 11th Iberian Conference on Information Systems and Technologies (CISTI)*. Las Palmas, Spain, 1-5.
3. Onan, A.; Korukoglu, S.; and Bulut, H. (2016). Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with* Applications, 57, 232-247.
4. Ageev, M.S.; and Dobrov, B.V. (2003). Support vector machine parameter optimisation for text categorization problems. *Proceedings of the International Conference on Information Systems Technology and its Applications (ISTA).* Kharkiv, Ukraine, 165-176.
5. Ilhan, I.; and Tezel, G. (2013). A genetic algorithm–support vector machine method with parameter optimisation for selecting the tag SNPs. *Journal of Biomedical Informatics*, 46(2), 328-340.
6. Ben-Hur, A.; Ong, C.S.; Sonnenburg, S.; Scholkopf, B.; and Ratsch, G. (2008). Support vector machines and kernels for computational biology. *PLoS Computational Biology*, 4(10), 10 pages.
7. Sharif, W.; Samsudin, N.A.; Deris, M.M.; and Naseem, R. (2016). Effect of negation in sentiment analysis. *Proceedings of the Sixth International IEEE Conference IEEE on Innovative Computing Technology (INTECH)*. Dublin, Ireland, 718-723.
8. Bothe, S.; Gartner, T.; and Wrobel, S. (2010). On-line handwriting recognition with parallelized machine learning algorithms. *Proceedings of the 33rd Annual German Conference on Advances in Artificial Intelligence.* Berlin, Heidelberg, 82-90.
9. Chen, H.; Jiang, W.; Li, C.; and Li, R. (2013). A heuristic feature selection approach for text categorization by using chaos optimisation and genetic algorithm. *Mathematical Problems in Engineering*, Article ID 524017, 6 pages.
10. Khan, A.; Baharudin, B.; Lee, L.H.; and Khan, K. (2010). A review of machine learning algorithms for text-documents classification. *Journal of Advances in Information Technology*, 1(1), 4-20.
11. Sharif, W.; Samsudin, N.A.; Deris, M.M.; and Aamir, M. (2017). Improved relative discriminative criterion feature ranking technique for text classification. *International Journal of Artificial Intelligence,* 15(2), 61-78.
12. Gaspar, P.; Carbonell, J.; and Oliveira, J.L. (2012). On the parameter optimisation of support vector machines for binary classification. *Journal of Integrative Bioinformatics*, 9(3), 11 pages.

13. Garsva, G.; and Danenas, P. (2014). Particle swarm optimisation for linear support vector machines based classifier selection. *Nonlinear Analysis: Modelling and Control*, 19(1), 26-42.
14. Yang, X.-S.; and Deb, S. (2009). Cuckoo search via levy flights. *Proceedings of the World Congress on Nature and Biologically Inspired Computing (NaBIC).* Coimbatore, India, 210-214.
15. Eberhart, R.; and Kennedy, J. (1995). A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science.* Nagoya, Japan, 39-43.
16. Aghdam, M.H.; and Heidari, S. (2015). Feature selection using particle swarm optimisation in text categorization. *Journal of Artificial Intelligence and Soft Computing Research,* 5(4), 231-238.
17. Mohapatra, P.; Chakravarty, S.; and Dash, P.K. (2015). An improved cuckoo search based extreme learning machine for medical data classification. *Swarm and Evolutionary Computation,* 24, 25-49.
18. Saadi, Y.; Yanto, I.T.R.; Herawan, T.; Balakrishnan.; V.; Chiroma, H.; and Risnumawan, A. (2016). Ringed seal search for global optimisation via a sensitive search model. *PloS One,* 11(1), 1-31.
19. Zong, W.; Wu, F.; Chu, L.-K.; and Sculli, D. (2015). A discriminative and semantic feature selection method for text categorization. *International Journal of Production Economics,* 165, 215-222.
20. Chen, Y.-W.; and C.J, Lin (2006). Combining SVMs with various feature selection strategies. *Feature Extraction,* Chapter 12, 315-324.
21. Allahyari, M.; Pouriyeh, S.; Assefi, M.; Safaei, S.; Trippe, E.D.; Gutierrez, J.B.; and Kochut, K. (2017). A brief survey of text mining: Classification, clustering and extraction techniques. *Proceedings of the KDD Bigdas.* Halifax, Nova Scotia, Canada, 1-13.
22. Tang, L.; and Liu, H. (2005). Bias analysis in text classification for highly skewed data. *Proceedings of the Fifth IEEE International Conference on Data Mining.* Houston, Texas, United States of America, 1-4.
23. Sharif, W.; Samsudin, N.A.; Deris, M.M.; and Khalid, S.K.A. (2017). A technical study on feature ranking techniques and classification algorithms.*Journal of Engineering and Applied Sciences,* 13(9), 7074-7080.