BUILDING SYSTEMATIC REVIEWS USING ONE-CLASS SVM AND FOOTBALL OPTIMIZATION ALGORITHM

Payam Hatamzadeh[1], Arash Tabe[2], Seyyed Keyvan Mousavi[3], Kaveh Shaker[4]

[1]Department of Computer Engineering, Faculty of Engineering,
University of Isfahan, Isfahan, Iran
[2]Computer Engineering Student, Department of Computer,
Tabriz University Campuses, Tabriz, Iran
[3]PhD Student, Department of Computer, Urmia Branch,
Islamic Azad University, Urmia, Iran
[4]Computer Engineering Department, Mizan Higher Education Institute, Tabriz, Iran

ABSTRACT

This paper describes a new approach for making automatic review systems, and shows how the football optimization can be used to improve the accuracy of one-class support vector machine. There are several challenges in building an automatic review system. The first challenge is the training samples being unbalanced. Another challenge is no access to the full texts of non-target classes. Applying one-class classification to learn from an unbalanced data set is regarded as the recognition-based learning and has shown to have the potential of achieving a better performance. Similar to two-class learning, parameter selection is a significant issue, especially when the classifier is sensitive to the parameters. This paper proposes the use of football optimization inspired by the football match to calibrate the parameters of one-class support vector machine for building an automatic review system. Like other evolutionary ones, the proposed algorithm starts with an initial population called team. Population individuals called players. Teamwork among these players forms the basis of the proposed evolutionary algorithm. Experiments performed on standard datasets show that the proposed method is able to match or surpass the performance of a one-class support vector machine with parameters optimized.

KEYWORDS

Automatic Review System, Unbalanced Samples, One-Class Support Vector Machine, Football Optimization

1. INTRODUCTION

In recent years, texts have widely been considered a source of information, so that knowledge discovery from texts is one of the most important activities that are known as text mining [1]. Text mining is described as the process of identifying novel information from a collection of texts. In this area, many activities and researches have been done, but the range of scope is very extensive. These attractions and activities are for high volume automatic processing of information.

One of the areas considered in this context is automatic review systems [2]. A systematic review is a literature review for identifying high quality research evidence relevant to specific research topic [3]. There are several challenges in building automatic review systems. The first challenge

is unbalanced training samples [4]. Applying conventional discrimination-based learning to unbalanced data, the classifier would easily have a bias towards the majority's class [5-7]. Thus, accuracy in classification minor categories will be less. Another challenge in this context is no access to the full texts of the irrelevant class. One-class classification, on the other hand, only uses the positive examples to train the classifier, so we do not need to worry about the diversity and the completeness of the negative class [8,9]. One of effective methods to solve the challenges mentioned is using one-class SVM.

An important thing in one-class SVM is high sensitivity to adjustable parameters and kernel function. Incorrect setting of these parameters reduces the accuracy algorithms on tested samples. In this paper, a novel technique is proposed for calibrate the parameters of one-class SVM inspired by the football game. The proposed algorithm starts with an initial population called team. A team composed of good passers and mobile players. All the players are divided into two types: main players and substitute players. Teamwork among main players is the main part of the proposed algorithm. During this Cooperation, main players pass the ball together and moving into free space. In addition, they move randomly by spectators' effect and per iteration replace weakest main players with best substitute players. This cycle is repeated until a termination condition has been reached.

The remainder of this paper is structured as follows. Section 2 briefly presents the relevant works done previously. Section 3 briefly introduces the basic definitions of the one-class SVM. Section 4 introduce the proposed algorithm and study its different parts in details. The proposed algorithm is tested with standard dataset in section 5 and section 6 concludes the paper.

2. RELATED WORK

A systematic review is a literature review for identifying high quality research evidence relevant to specific research topic. Most such systems are used to recover certain articles in the same fields. The research done by [10] is probably the first application of automatic text classification to the task of creating systematic reviews. In this paper, the authors experimented with medical articles. Further work for systematic reviews was done [11] where text classification was used for automatic review systems. Their work is mostly focused on the use of feature selection algorithms and filtering techniques in text classification system. In [12] machine-learning technology driven by human determination was used to build text review system. In [13] text classification system was used to build systematic reviews too. This paper presents a novel text classifier from positive and unlabelled documents based on GA. [14] proposed the bagged one-class SVM algorithm as opposed to the one-class SVM. Although, bagging one-class SVM improved the decision boundary on the 2-D synthetic examples, there was no noticeable improvement on the real world data sets, which are much more representative of those that would be encountered in practice.

On the other hand, one-class SVM was first proposed in [15] to estimate the probability density function where the data set is drawn from. This algorithm was used when access to a non-target class is expensive or impossible. Empirical results in [16] show that on heavily unbalanced data, one-class classifier achieves much better performance than the conventional two-class ones. In this paper, one-class SVM algorithm in the some cases mentioned is more effective than SVM. When negative classes are completeness and when samples are unbalanced. One-class SVM is one of the active research focuses. It has been applied to document classification in [17].

3. BACKGROUND

One-class SVM algorithm is used for classification and clustering [18]. This aims to solve this problem [15]:

"Suppose a data set is drawn from an underlying probability distribution P. Estimate a "simple" subset Sin the input space such that the probability that a test point drawn from Plies outside of S equals some priori specified value between 0 and 1.

In short, this algorithm has two stages as follows:

1-Map the data into higher dimension space and corresponding with kernel function.

2- Separate the mapped vectors from the origin with maximum margin.

Let x_i denote as positive training samples and $\varphi : X \to H$ be a kernel map, which transforms the data into an inner product space H. The problem of separating the data set from the origin is essentially the problem of optimizing the following quadratic programming:

$$\min\left\{ \tfrac{1}{2}\|w\|^2 + \tfrac{1}{lv}\sum_{i=1}^{l}\varepsilon_i - \rho \right\}$$

$$s.t. \quad v \in (0,1) \quad , \quad i = 1,2,\dots,l \tag{1}$$

$$(w.\varphi(x_i)) \geq \rho - \varepsilon_i \quad , \quad \varepsilon_i \geq 0$$

Where ε_i are so-called slack variables that penalize the objective function but allow some of the points to be on the wrong side of the hyperplane, i.e. located between the origin and $H(w,\rho)$ as depicted in Figure 1. l is number of training samples and $v \in (0,1)$ is a parameter that controls the tradeoff between maximizing the distance from the origin and containing most of data in the region created by hyperplane.
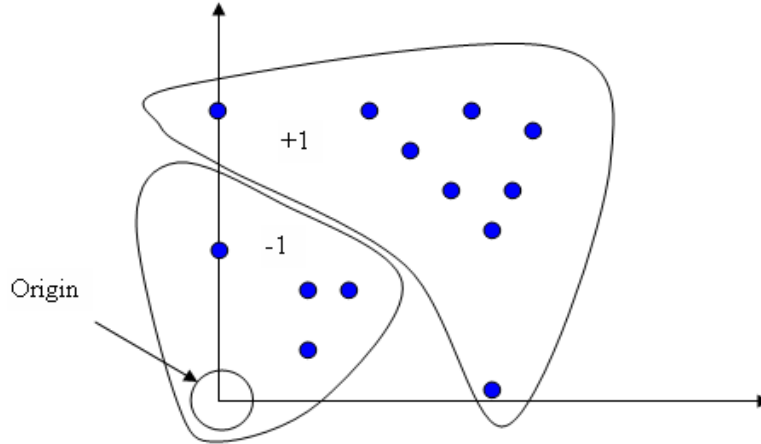

Figure 1. One-class Support Vector Machines in two-dimensional feature space

4. THE PROPOSED ALGORITHM

Figure 2 shows the flowchart of the proposed algorithm. FO algorithm encodes a potential solution to text classifiers on players and applies cooperation operators to these players. This algorithm is viewed as a function optimizer although the range of problems to which this algorithm has been applied to, is quite broad. Training is done by using one-class support vector machines and samples of target class. At this stage speed is high because the one class classifier is constructed using the minority class.
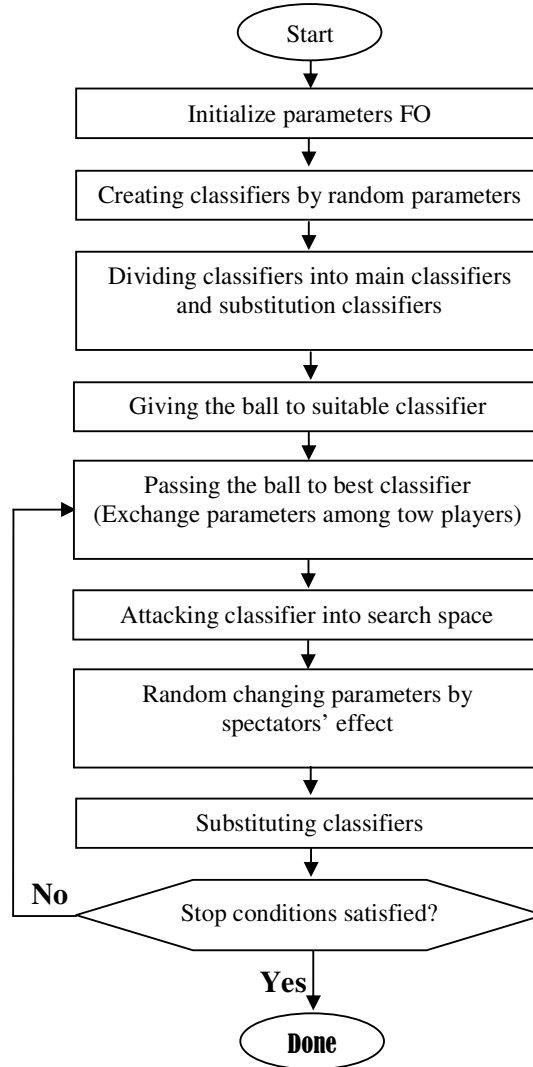
```
                          ┌─────────┐
                          │  Start  │
                          └────┬────┘
                               ▼
            ┌──────────────────────────────────────┐
            │       Initialize parameters FO        │
            └──────────────────┬───────────────────┘
                               ▼
            ┌──────────────────────────────────────┐
            │  Creating classifiers by random       │
            │             parameters                │
            └──────────────────┬───────────────────┘
                               ▼
            ┌──────────────────────────────────────┐
            │ Dividing classifiers into main        │
            │ classifiers and substitution          │
            │ classifiers                           │
            └──────────────────┬───────────────────┘
                               ▼
            ┌──────────────────────────────────────┐
            │  Giving the ball to suitable          │
            │          classifier                   │
            └──────────────────┬───────────────────┘
                               ▼
            ┌──────────────────────────────────────┐
            │ Passing the ball to best classifier   │
            │ (Exchange parameters among tow        │
            │  players)                             │
            └──────────────────┬───────────────────┘
                               ▼
            ┌──────────────────────────────────────┐
            │ Attacking classifier into search      │
            │            space                      │
            └──────────────────┬───────────────────┘
                               ▼
            ┌──────────────────────────────────────┐
            │ Random changing parameters by         │
            │       spectators' effect              │
            └──────────────────┬───────────────────┘
                               ▼
            ┌──────────────────────────────────────┐
            │       Substituting classifiers        │
            └──────────────────┬───────────────────┘
                               ▼
    No      ◇─────────────────────────────────────◇
  ◄─────────  Stop conditions satisfied?
            ◇─────────────────────────────────────◇
                           Yes │
                               ▼
                          ┌─────────┐
                          │  Done   │
                          └─────────┘
```

Figure 2. Flowchart of the proposed algorithm

4.1. Initialize parameters

Table 1 shows the adjustable parameters of the football optimization algorithm.

Table 1. Adjustable parameters of the FO algorithm

Parameter	Description	Value
n	Maximum number of players	$[2,\infty)$
α	Divide coefficient of players	$(0,1]$
θ	Number of replacement in entire iterations	$[0,n]$
δ	Number of replacement per iteration	$[0, n\times \alpha]$
β	Pass coefficient	$[0,1]$
γ	Velocity coefficient of players	best value $[0.5,2]$
λ	Spectators' effect on players	$[0,1]$
σ	Spectators' effect on parameters	$[0,1]$

4.2. Creating classifiers by random parameters

The first step in the implementation of any optimization algorithm is to generate an initial population. In a FO algorithm, a population of players called Team, which encode candidate solutions to an optimization problem, evolves toward better solutions. In other words, each player creates by $1 \times k$ array. Table 2 shows the adjustable parameters of the one-class SVM algorithm.

Table 2. Adjustable parameters of the one-class SVM algorithm

Parameters	Value
Kernel type	0 -- linear: u'*v 1 -- polynomial: (gamma*u'*v + coef0)^degree 2 -- radial basis function: exp(-gamma*|u-v|^2) 3 -- sigmoid: tanh(gamma*u'*v + coef0)
degree	set degree in kernel function (default 3)
gamma	set gamma in kernel function (default 1/k)
coef0	set coef0 in kernel function (default 0)
v	Fraction of outliers and support vectors
ε	Slack Variables

The population size (n) depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. This algorithm usually starts from a population of randomly generated individuals and covers the entire range of possible solutions (the search space).

$$Classifier = \{KernelType, \deg ree, gamma, coef0, v, \varepsilon\} \tag{2}$$

$$Team = \begin{bmatrix} Classifier_1 \\ Classifier_2 \\ \vdots \\ Classifier_n \end{bmatrix}_{n \times k} \tag{3}$$

4.3. Dividing classifiers into main classifiers and substitution classifiers

Before the games, coaches always trained the team because it can play an important part in a match. After practice players, they selected most powerful them to form main team. After training classifiers, m of the most powerful classifiers are selected to form the main players. The remaining of the population (s) will be remaining as reserve players. Then, we have two types of players: main players and reserve players.

$$mainPlayers = \begin{bmatrix} Classifier_1 \\ \vdots \\ Classifier_m \end{bmatrix}_{m \times k}$$

$$substitutePlayers = \begin{bmatrix} Classifier_{m+1} \\ \vdots \\ Classifier_n \end{bmatrix}_{s \times k} \tag{4}$$

$$where \quad m = round(n \times \alpha) \quad, s = n - m$$

4.4. Giving the ball to suitable classifier

From the beginning of each playing period until the end of the playing period, there is one ball in football game. Hence, one classifier of the existing main players is selected to get possession of

of the ball. An individual classifier is selected through a performance measure based process, where fitter classifiers (as measured by a performance measure) are typically more likely to be selected. Other methods rate only a random sample of the classifiers, as this process may be very time-consuming. Equation 5 determines the certain selection method that rates the performance measure of each classifier, add random value (with uniform distribution) to it, and preferentially select the best rank classifier.

$$IndexOwnerBall = IndexBest\{RankClassifier_1,\dots,RankClassifier_m\}$$
$$RankClassifier_i = Fitness(Classifier_i) + U(-d,+d)$$

(5)

4.5. Passing the ball to the best classifier (Exchange parameters among tow players)

Passing the ball is a key part of football. The purpose of passing is to keep possession of the ball by maneuvering it on the ground between different players and to advance it up the playing field. Aside from having conspicuous advantages, passing is a skill that demands good technical ability not only from the distributor but from the receiver as well. In this algorithm, Passing is a tool with great creative potential and always has to be directed at a teammate's feet. The pass is considered as an offensive action. Thus upon figure 3, in each iteration, the performance measure of every classifiers in the population is evaluated, best-ranked classifiers are selected from the current main player (based on their fitness), and $\beta \times k$ parameters exchanged between passer and it.
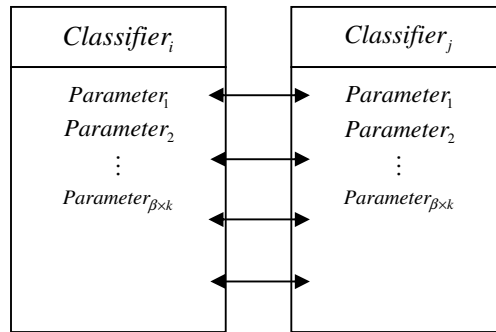

Figure 3. Exchange $\beta \times k$ parameters between two classifiers

4.6. Random changing parameters by spectators' effect

Once a player has passed the ball, other players do not remain stationary but move into a position where they can receive the ball and give more options to the player in possession. Moving into free spaces is one of the most critical skills that footballers must develop. Players must move off the ball into space to give an advance the maximum chance of success. Passes to space are feasible when there is intelligent movement of players to receive the ball and they do something constructive with it. In this algorithm, players move into the search space. The proposed algorithm has modeled this fact by moving all the classifiers toward the best classifier. This movement is shown in figure 3 in which the players move towards the best players by x units. The direction of the movement is the vector from each player to best player. In Equation 6, x is a random variable with uniform (or any) distribution, γ is a number greater than zero and causes the players to get closer to the goal from any side and d is the distance between the best player and other players.

$$if \quad d \geq 0 \quad Then \quad x \in (0,U(\gamma * d)) \quad else \quad x \in (U(\gamma * d),0)$$

(6)

Figure 4. Attack classifiers toward goal in proposed algorithm

4.7. Random changing parameters by spectators' effect

The impact of spectators upon sport is substantial and varied. Spectators are one of the reasons for the success of football teams. Spectators at the stadium and team practice increase morale and sense of responsibility in football players. This feeling will be transferred among all players and even coaches and managers. This movement is shown in figure 5 in which spectators' effect modeling by random change in classifiers parameters. In Equation 7, m is a number of main players and k is number of parameters of each classifier.

$$NumberOfEffectOnPlayers = m \times \lambda$$
$$NumberOfEffectOnParameters = k * \sigma$$

(7)
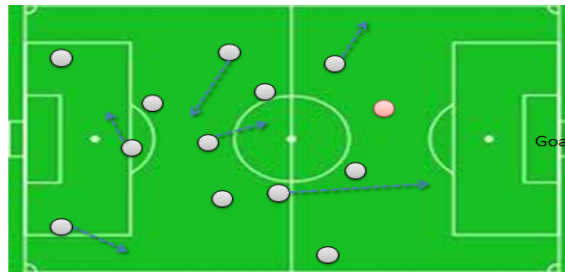
Figure 5. Random movement classifiers by spectators' effect in proposed algorithm

4.8. Substituting classifiers

A number of players may be replaced by substitutes during the course of the football game. Common reasons for a substitution include injury, tiredness, ineffectiveness, a tactical switch, or time wasting at the end of a finely poised game. The most tired players are generally substituted, but coaches often replace ineffective players in order to freshen up the attacking posture and increase their chances of scoring. In this algorithm, like football matches, substituting players is required to make the conditions better. This can vary during the game and put a significant impact on the success of the team. The number of substitutes must be determined before the algorithm begins, which may be anywhere between zero and n. Thus in each iteration, the fitness of every classifiers in the team is evaluated and a comparison between the weakest main player and the best substitute player takes place. If the substitute player is stronger than the main player, a switch takes place. During this execution algorithm can use of θ (number of replacement) to adjust parameters. For example if θ is very high must decrease ε (spectators' effect on players).

4.9. Initialize parameters

This process is repeated until a termination condition has been reached. Common terminating conditions are:

- A solution is found that satisfies minimum criteria (min (1- Performance Measure)).
- Fixed number of iterations reached.
- Allocated budget (computation time/money) reached.
- The highest solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results.
- Manual inspection.
- Combinations of the above.

5. EXPERIMENTAL RESULTS

One-Class SVM is included in the LIBSVM software package [19]. LIBSVM is an integrated tool for support vector classification and regression. Based on the library provided, the proposed framework implemented in C#.

5.1. Performance Measure

To evaluate the performance of the proposed system, several evaluation metrics are considered: accuracy, g-means, and F-measure. Despite the classified text output as O= {+1, -1}, Table 3 shows the number of samples classified based on output.

Table 3. Number of samples classified based on predicted output and real output

		Predicted output		
		+1	-1	Σ
Real output	+1	N_{00}	N_{01}	$N_{0.}$
	-1	N_{10}	N_{11}	$N_{1.}$
	Σ	$N_{.0}$	$N_{.1}$	N

Accuracy is a common measure metric in text classification. It represents the percentage of documents that are correctly classified as follows:

$$M = \frac{N_{00} + N_{11}}{N} \tag{8}$$

However, when the class distribution is highly imbalanced, accuracy cannot measure the performance properly. G-means metric used as a common practice in the performance evaluation of algorithms in imbalanced data classification. G-means is denoted as:

$$G = \sqrt{sensitivity \times specificity}$$
$$G = \sqrt{\frac{N_{00}}{N_{0.}} \times \frac{N_{11}}{N_{1.}}} \tag{9}$$

Where sensitivity is the number of positive samples correctly classified to the total number of positive samples and specificity is the number of correctly classified negative samples to total number of negative samples. Harmonic Average or F1 measure used as a standard measure in

most papers on text categorization. The two components of the recall measure and precision are made as follows [17]:

$$F1 = \frac{2 \times recall \times precision}{recall + precision}$$

$$recall = \frac{N_{00}}{N_{0.}} \quad , precision = \frac{N_{00}}{N_{.0}}$$

(10)

5.2. Test Results

Experiments are comparing the standard algorithm and optimized algorithm with FO. In the experiments, used standard dataset commonly used in text classification tasks. These algorithms were applied to 20-Newsgroups data sets [20]. 20-Newsgroups contain texts, which collected from 20 UseNet groups with nearly 1000 texts from each group. All the 20 categories were used as target categories in these experiments. Thus, using one-against-all learning strategy and the imbalance ratio is about 19 for each category (the class distribution in this dataset is 5% in a relevant class and 95% in an irrelevant class). To train is used 100 texts in each category and all texts (about 20,000 texts) are used for testing. In standard classifier, initial setting is chosen as the default. Linear kernel function is used to its simplicity and effectiveness of a one-class SVM algorithm. The system evaluates the classifier performance by 100 texts in all categories, per iteration of algorithm.

In table 4, accuracy of target class and non-target class for both algorithms are shown separately. The important point in this table is the high accuracy non-target class to target class in standard algorithm (nearly 1.5 times). This difference is because the adjustable parameters are constant. Experiments performed in figure 6, on standard datasets show that the proposed method is able to match or surpass the performance of a one-class SVM with optimized parameters. This result shows its ability of optimizing algorithms for building automatic review systems.

Table 4. Performance results for standard algorithm and optimized algorithm

| Index | Categories | Precision | | | |
| | | Standard OC-SVM | | Optimized OC-SVM | |
		Target	Non target	Target	Non target
1	alt.atheism	0.514	0.801	0.88	0.709
2	comp.graphics	0.539	0.763	0.773	0.701
3	comp.os.ms-windows.misc	0.525	0.844	0.884	0.71
4	comp.sys.ibm.pc.hardware	0.487	0.68	0.702	0.6
5	comp.sys.mac.hardware	0.527	0.841	0.861	0.75
6	comp.windows.x	0.553	0.734	0.754	**0.74**
7	misc.forsale	0.509	0.738	0.788	0.699
8	rec.autos	0.477	0.715	0.715	0.697
9	rec.motorcycles	0.426	0.624	0.664	0.606
10	rec.sport.baseball	0.503	0.691	0.693	0.603
11	rec.sport.hockey	0.505	0.777	0.767	0.755
12	sci.crypt	0.502	0.749	0.802	0.703
13	sci.electronics	0.5	0.645	0.635	**0.69**
14	sci.med	0.386	0.73	0.909	0.71
15	sci.space	0.484	0.707	0.737	0.68
16	soc.religion.christian	0.377	0.951	0.991	0.84
17	talk.politics.guns	0.589	0.808	0.808	0.75
18	talk.politics.mideast	0.47	0.832	0.852	0.78
19	talk.politics.misc	0.575	0.699	0.706	0.633
20	talk.religion.misc	0.462	0.705	0.795	0.65
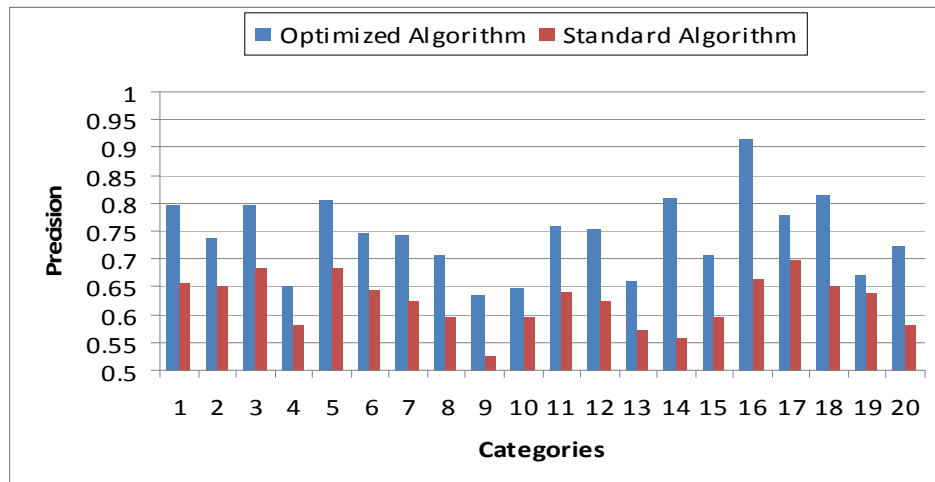
Figure 6. Mean accuracy for standard algorithm and optimized algorithm

5. CONCLUSIONS AND FUTURE WORK

In the traditional Systematic Review System, reviewers or domain experts manually classify literatures into a relevant class and an irrelevant class through a series of systematic review levels. This process is very time consuming and costly and therefore restricts its application. Therefore, increasing interest is expressed to the development of automated technologies. In this paper, an optimization one-class SVM based on modelling the football game is proposed. In FO, each individual of the population is called players. The team is divided into two groups: main players and reserve players. The teamwork among main players forms the core of this algorithm and results in the convergence of ball to the goal as expected. In this cooperation, the ball is moved gradually to the goal and finally best classifier is selected. The algorithm is tested by benchmark datasets and this is compared with a standard one-class SVM. The results show that the proposed algorithm increases the accuracy. Future work will consist in modifying some parts of the algorithm improve the algorithm execution speed.

REFERENCES

[1] A.K. Sehgal, (2004) "Text Mining: The Search for Novelty in Text," Dept. of Computer Science, University of Iowa, Ph.D. Comprehensive Examination Rep. ,.

[2] S. Ananiadou, R. Procter, B. Rea, Y. Sasaki, J. Thomas, (2009) "Supporting Systematic Reviews Using Text Mining," Social Science Computer Review Journal, vol. 27, Issue 4, pp. 509-523.

[3] Tsafnat G, Glasziou P, Choong MK, Dunn A, Galgani F, Coiera E. (2014) "Systematic review automation technologies. Systematic Reviews", DOI: 10.1186/2046-4053-3-74

[4] Y. Chen, (2008) "Learning Classifiers from Imbalanced, Only Positive and Unlabeled Data Sets," in 14th ACM SIGKDD international conference on Knowledge discovery and data mining, USA, pp. 213-220.

[5] A. Sun, E.P. Lim, Y. Liu, (2009) "On Strategies for Imbalanced Text Classification Using SVM: A Comparative Study," Decision Support Systems, vol. 48, Issue 1, pp. 191-201, Dec.

[6] L. Zhuang, H. Dai, (2006) "Parameter Optimization of Kernel-based One-class Classifier on Imbalance Learning," Trends in Artificial Intelligence, vol. 4099, pp. 434-443.

[7] M. Hovelynck, (2009) "Multi-modal Classification in a One-class Setting," M.S. thesis, Dept. Tech. Eng. , Utrecht Univ. , Utrecht.

[8] S. R. Jonnalagadda, P. Goyal and M. D. Huffman, (2015) "Automating data extraction in systematic reviews: a systematic review. Systematic Reviews", DOI: 10.1186/s13643-015-0066-7

[9] Adeva JJG, Pikatza Atxa JM, Ubeda Carrillo M, Ansuategi ZE. (2014) "Automatic text classification to support systematic reviews in medicine. Expert Systems and Applications", 41(4):1498-1508

[10] W. Bartling, T. Schleyer, S. Visweswaran, (2003) "Retrieval and Classification of Dental Research Articles," Advances in Dental research.

[11] Y. Ma, (2007) "Text classification on imbalanced data: Application to Systematic Reviews Automation," M.S. thesis, Dept. Tech. Eng. , Ottawa Univ. , Ottawa, Canada.

[12] T. Barnett, S. Godjevac, J.M. Renders, C. Privault, J. Schneider, R. Wickstorm, (2009) "Machine Learning Classification for Document Review," in 12th International Conference on Artificial Intelligence & Law, Barcelona Spain.

[13] O. Frunza, D. Inkpen, S. Matwin, (2010) "Building Systematic Reviews Using Automatic Text Classification Techniques," in Proceedings of the 23th International Conference on Computational Linguistics, pp. 303–311.

[14] D. Kamm, (2008) "Bagging for One-Class Learning," in 229th conference Machine Learning, Pittsburgh Univ. , USA.

[15] B. Scholkopf, J.C. Platt, J.S. Taylor, A.J. Smola, R.C. Williamson, (2001) "Estimating the Support of a High-Dimensional Distribution," Technical report of Microsoft Research, vol. 13, Issue 7.

[16] O'Mara-Eves A, Thomas J, McNaught J, Miwa M, Ananiadou S. , (2015) "Using text mining for study identification in systematic reviews: a systematic review of current approaches". doi:10.1186/2046-4053-4-5.

[17] L.M. Manevitz, M. Yousef, "One-class SVM for document classification," Journal of Machine Learning Research, vol. 2, pp. 139-154, Mar. 2001.

[18] G. Hong, Z. Guangzhou, Q. Jun, (2010) "One-Class Support Vector Machine with Relative Comparisons," Tsinghua Science and Technology, vol. 15, num. 2, pp. 190-197.

[19] C.C. Chang, C.J. Lin, (2006) "LIBSVM: a Library for Support Vector Machines", Software Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[20] Newsgroups dataset (1999, Sep. 9). Available at: http://kdd.ics.uci.edu/databases/20newsgroups.