Modified Multi-Class Miner using Particle of Swarm Optimization for Stream Data Classification

Archana Bopche
M. Tech. scholar
PIES, Bhopal, India

Parmalik Kumar
Department of Computer science & Engineering
PCST, Bhopal, India

ABSTRACT
Multi-class miner is well recognized method for stream data classification. For the process of multi-class miner evaluation of new feature during classification is major problem. The problem of feature evaluation decreases the performance of multi-class miner (MCM). For the improvement of multi-class miner particle of swarm optimization technique is used. Particle of swarm optimization controls the dynamic feature evaluation process and decreases the possibility of confusion in selection of class and increase the classification ratio of multi-class miner. Particle of swarm optimization work in two phases one used as dynamic population selection and another are used for optimization process of evolved new feature. For the performance evaluation modified MCM algorithm implemented in MATLAB. For the validation of modified multi-class miner (MMCM) used sample dataset from UCI machine learning repository .Our empirical evaluation shows that better result in compression of multi-class miner and also increases the classification ratio of stream data classification.

General Terms
Stream data classification, Swarm Intelligence

Keywords
Stream Data, MCM, POS.

1. INTRODUCTION
The increasing rate of internet multimedia data and sensor data of forest put high demand of valid classification technique of stream data [1]. The technique improves the classification performance of stream data. In current age of technology of stream data classification various data mining algorithm are available. Some algorithm supports vector clustering, support vector machine, multi-class miner and DXminer. These entire algorithms not resolve all problem of multi-class miner such as infinite length, feature evaluation, concept evaluation and data drift. The problem concern such are feature evaluation and data drift are major part in concern of classification. In this regards different authors used optimization algorithm such as genetic algorithm, neural network technique and ant colony optimization. The aim of data stream investigation is to make decisions based on the outline in sequence gathered over the past experiential data essentials. Regular techniques contain classification, clustering, model ensemble, multiple or distributed data stream clustering, change diagnosis, query processing, etc. Among them, clustering is one of the most effective means of summarizing data streams and building a model for visualization and analysis [6]. Traditional stream classification techniques also make impractical assumptions about the availability of labeled data. Most techniques assume that the true label of a data point can be accessed as soon as it has been classified by the classification model. Thus, according to their postulation, the existing model can be updated without delay using the labeled instance. In reality, one would not be so lucky in obtaining the label of a data instance immediately, since manual labeling of data is time consuming and costly. We claim two major contributions in novel class detection for data streams [7]. First, we propose a dynamic selection of boundary for outlier detection by allowing a slack space outer the decision boundary. This space is restricted by a threshold, and the threshold is modified all the time to reduce the risk of false alarms and missed novel classes. Subsequent, a probabilistic approach to detect novel class instances using the local search of particle of swarm optimization is being affected. With this approach, one is able to distinguish different causes for the appearance of the outliers, specifically concept-drift, concept-evolution or noise. The contributions of this work are as follows. First, this work addresses the dynamic class issue and concept-evolution in the data streams. Here proposed solution, which uses particle of swarm optimization for class detection, reduces false alarm rates and overall classification error. Second, this technique can be applied to detect periodic classes, such as classes that appear tabloid, monthly, or yearly [4]. This will be useful for better predicting and profiling the characteristics of a data stream. Finally, we apply our technique on a number of real and synthetic datasets, and obtain superior performance over state-of-the-art techniques. Swarm Intelligence is a relatively new interdisciplinary field of research; this has gained huge popularity in these days. Algorithms belonging to colonies without any kind of supervisor or controller. Particle Swarm Optimization (PSO) is another very popular SI algorithm for global optimization over continuous search spaces. PSO has attracted the attention of several researchers all over the world resulting into a huge number of variants of the basic algorithm as well as many parameter automation strategies. The above section discuss introduction of stream data classification and feature optimization. In section II multi-class miner and particle of swarm optimization is described. In section III discuss proposed method. In section IV experimental process and finally concluded in section V.

2. MULTI-CLASS MINER AND POS
In this section describe two algorithms one is multi-class miner and other one is particle of swarm optimization. The first phase of algorithm discuss multi-class miner algorithm and second phase discuss particle of swarm optimization algorithm.

Phase-I
The multi-class miner algorithm is a combination of clustering and classification technique such technique also called ensemble process [3]. The main idea in detecting multiple novel classes is to build a graph, and discover the connected components in the graph. The number of connected

components regulates the number of novel classes. The essential postulation in formative the multiple novel classes' follows property: A data point should be closer to the data points of its own class (cohesion) and farther apart from the data points of other classes (separation). If there is a novel class in the stream, instances. For example, if there are two novel classes, then the partition among the dissimilar novel class instances should be higher than the cohesion among the same-class instances.

Input: N_list: listing of novel class instances
Output: N_type: anticipated class label of the novel instances

1: G = (V, E) ←empty //initialize graph
2: NP_list←K-means(N_list, K_v) //clustering
3: for h ∈ N P_list do
4: h.nn ← Nearest-neighbor (NP_list - {h})
5: h.sc←Compute-SC(h,h.nn) //silhouette coefficient
6: V←V ∪ {h} //add these nodes
7: V←V ∪{h.nn}
8: if h.sc < th_{sc} then //relatively closer to the nearest neighbor
9: E←E ∪ {(h,h.nn)} //add this directed edge
10: endif
11: end for
12: count ← Con-Components (G) //find connected components // Merging phase
13: for each pair of components (g1,g2) ∈ G do
14: μ_1←mean-dist (g1), μ_2←mean-dist (g2)
15: If $\frac{\mu_1+\mu_2}{2*\text{centroid\_dist(g1,g2)}} > 1$ then g1←Merge (g1, g2)
16: end for // Now allot the class labels
17: N_type ← empty
18: for x ∈ N list do
19: h ← PseudopointOf (x) //find the corresponding pseudopoint
20: N_type ← N_type ∪ {((x) , h.ccomponent)}
21: end for

Phase-II
In Particle Swarm Optimization [10] optimizes an objective function by undertaking a population based search. The population comprise of possible solutions, named particles, which are metaphor of birds in flocks. These particles are at random initialized and freely fly across the multi dimensional seek space. During flight, each particle updates its own velocity and position based on the best experience of its own and the entire population. The different steps involved in Particle Swarm Optimization Algorithm are as follows:

Step 1: All particles' velocity and position are randomly place to within pre-defined ranges.

Step 2: Velocity update – At every iteration, the velocities of all particles are updated based on below expression

$$v_i = v_i + c_1 R_1 (p_{i,best} - p_i) + c_2 R_2 (g_{i,best} - p_i) \quad ....(1)$$

where p_i is the position and v_i are the velocity of particle i, $p_{i,best}$ and $g_{i,best}$ is the position with the 'best' objective value found so far by particle i and the entire population respectively; w is a parameter controlling the dynamics of flying; R_1 and R_2 are random variables in the range [0,1]; c_1 and c_2 are factors controlling the related weighting of equivalent terms. The random variables facilitate the PSO with the ability of stochastic searching.

Step 3: Position updating – The positions of all particles are updated according to,

$$p_i = p_i + v_i \quad ...(2)$$

Following updating, p_i should be verified and limited to the allowed range.

Step 4: Memory updating – Update $p_{i,best}$ and $g_{i,best}$ when condition is met,

$$p_{i,best} = p_i \quad if\ f(p_i) > f(p_{i,best})$$

$$g_{i,best} = g_i \quad if\ f(g_i) > f(g_{i,best}) \quad ...(3)$$

Where $f(x)$ is to be optimized and it is a objective function.
Step 5: Stopping Condition – The algorithm repeats steps 2 to 4 until certain stopping circumstances are met, such as a pre-defined number of iterations. Once closed, the algorithm reports the values of g_{best} and $f(g_{best})$ as its solution[8].

PSO utilizes several searching points and the searching points gradually get close to the global optimal point using its pbest and gbest. Primary positions of pbest and gbest are dissimilar However, using thee different direction of pbest and gbest, all agents progressively get close up to the global optimum.

3. MODIFIED MULTICLASS MINER (MMCM-POS)

In this section we discuss the modification of multi-class miner algorithm with particle of swarm optimization. POS is heuristic function; the nature of POS is multi-objective for optimization of given problem. In multi-class miner POS plays a role of seed selection of better generation of cluster radius for grouping of new feature data in ensemble process. In the process of MCM the graph points of number of feature point selection executed by particle of swarm optimization [9]. Modified Steps of MCM-POS algorithm Input: N_list: Listing of novel class instance Output: N_type: anticipated class label of the novel instances

1: G = (V, E) ←empty //initialize graph
2: NP_list ← K-means (N_list, K_v)
3: Input NP_list X , the clustering number pn , population scale XN ,velocity probability vP stop conditions cS ;
4: Code the particle in real number and initialize population S(i),i = 0 at random;
5: Evaluate the fitness of all individual in the current instant D(s);
6: MCM clustering requires optimization of cluster center, which way thrashing of data of waiting cluster. Hence the fitness function of algorithm is determined by f(x).
7: $G(s) = \frac{N(s)}{D(s)} = \frac{\sum_{i=0}^{n-1} A_i s^i}{\sum_{i=0}^{n} a_i s^i}$ Umpire the termination conditions. If the termination situation are satisfied, then turn to step 9, if not, turn to step 10;

8: Crack to find and compute the optimal clustering centers.

9: find final population of POS

10: Take the MCM optimization on population P(i) and generate the next generation A(i +1) . Then turn to step

11: for h ∈ A(i+1) do
12: h.nn ← Nearest-neighbor (A(i+1)- {h})
13: h.sc ← Compute-SC (h, h.nn)
14: V←V ∪{h}
15: V←V ∪{h.nn}
16: if h.sc < th_{sc} then
17: E←E ∪ {(h,h.nn)}
18: endif
19: end for
20: count ← Con-Components (G)
for each pair of components (g1,g2) ∈ G do
21: μ_1←mean-dist (g1), μ_2←mean-dist (g2)
22: if $\frac{\mu_1+\mu_2}{2*centroid\_dist(g1,g2)} > 1$ then g1←Merge (g1, g2)
23: end for
// Now assign the class labels
24: N_type ← empty

25: for x ∈ Nlist do
26: h←PseudopointOf(x)//find the corresponding pseudopoint

27: N_type ←N_type∪{(x , h.componentno)}
28: end for

4. EXPERIMENTAL PROCESS

For the evaluation of performance of MCM and MCM-POS, we implement our algorithm in mat lab 7.8.0 and for tested of result used UCI machine repository data set. Here three data set glass data set, crop data set and finally forest fire data are used. The result measurement parameter is Fnew , Mnew and Error rate of classification. Evaluation table of result are shown below.

In the table below result of both methods MCM and MCM-POS computed for crops data set.

Table 1 Calculated result of feature selection of MCM and MCM-POS for crops data set are shown

Chunk size	Error rate	Mnew	Fnew	Time in seconds
10	**0.569**	**13.5 34**	**0.4373**	**0.328215**
20	**0.769**	**12.786**	**0.5216**	**0.453076**
30	**0.989**	**11.368**	**0.3426**	**0.631141**
40	**0.536**	**15.865**	**0.3688**	**0.743038**
50	**0.434**	**16.458**	**0.7586**	**0.562971**
60	**0.379**	**17.336**	**0.6587**	**0.860288**
70	**0.289**	**15.123**	**0.5689**	**0.730222**

Table 2 Shows computed result for both methods MCM and MCM-POS for crops data set

Chunk size	Error rate	Mnew	Fnew	Time in seconds
10	**0.458**	**5.164**	**12.601694**	**11.212107**
20	**0.428**	**4.563**	**14.228785**	**12.609094**
30	**0.364**	**5.224**	**14.329298**	**13.659516**
40	**0.338**	**4.144**	**18.324813**	**14.230065**
50	**0.278**	**5.298**	**28.463011**	**28.624513**
60	**0.122**	**6.132**	**31.367421**	**30.632463**
70	**0.102**	**5.456**	**51.797559**	**48.618276**

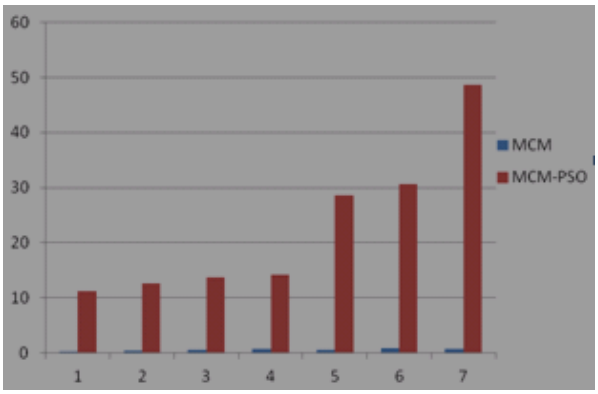

Figure 1: Comparison of error rate between MSM and MSM-PSO

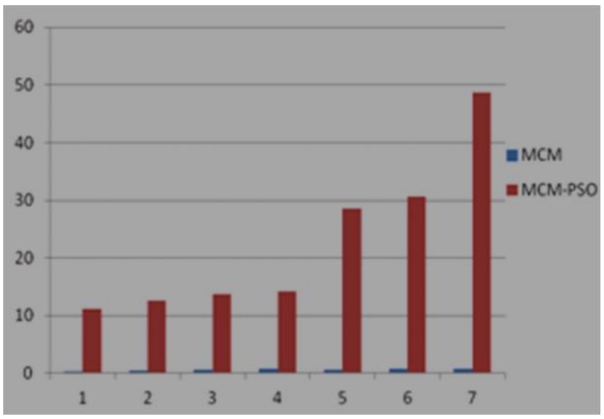

Figure 2: Comparison of M new between MCM and MCM-PSO

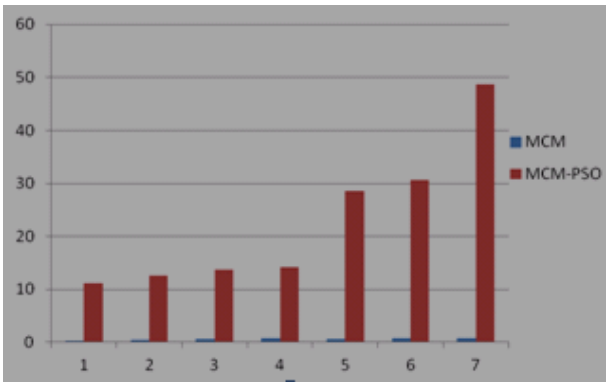

Figure 3: Comparison of F new between MCM and MCM-PSO

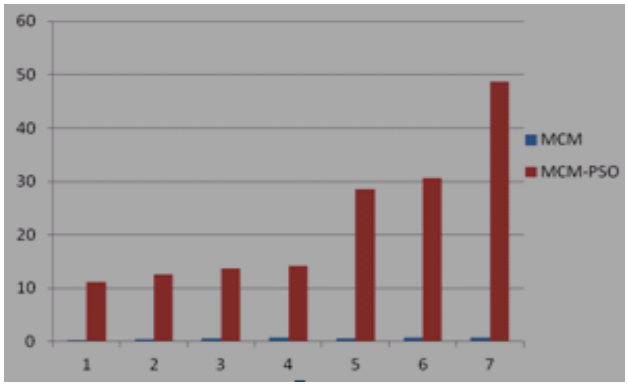

Figure 4: Comparison of elapse between MCM and MCM-PSO

5. CONCLUSION

Multi-class miner is very efficient data mining tool for stream data classification. Stream data classification is challenging task in the field of classification. Evaluation of new feature creates a problem in feature selection during the classification process of multi-class miner. In this paper reduces these problems using particle of swarm optimization, particle of swarm optimization used to control new feature evolution problem. Particle of swarm optimization creates a feature prototype for cluster used in classification. The empirical evaluation of modified algorithm is better in compression of MCM algorithm. The error rate of modified algorithm decreases in compression of MCM algorithm. Also improved the rate of Fnew and Mnew for evolution of result. the particle of swarm prototype cluster faced a problem of right number of cluster, in future used self optimal clustering technique along with particle of swarm optimization.

6. REFERENCES

[1] Urvesh Bhowan, Mark Johnston, Mengjie Zhang and Xin Yao "Evolving Diverse Ensembles using Genetic Programming for Classification with Unbalanced Data" in IEEE Tansaction2010.

[2] Yan-Nei Law and Carlo Zanily entitled" An Adaptive Nearest Neighbor Classification Algorithm for Data Streams" in PKDD 2005, LNAI 3721, pp. 108–120, 2005.

[3] Mohammad M. Masud, Latifur Khan, Bhavani Thuraisingham entilied "Classification And Novel Class Detection In Concept-Drifting Data Streams Under Time Constraints" in IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 23, NO. 6, JUNE 2011.

[4] Mohammad M. Masud, Qing Chen, Latifur Khan, Charu Aggarwal and Jing Gao , Jiawei Han and Bhavani Thuraisingham "Addressing Concept-Evolution in Concept-Drifting Data Streams " in IEEE Transaction 2010.

[5] Valerio Grossi, Alessandro Sperduti "Kernel-Based Selective Ensemble Learning for Streams of Trees" in Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence 2010.

[6] Li Su Xi, Hong-yan Liu, Zhen-Hui Song. "A New Classification Algorithm for Data Stream".

[7] Clay Woolam, Mohammad M. Masud, and Latifur Khan "Lacking Labels in the Stream: Classifying Evolving Stream Data with Few Labels" in *I.J.Modern Education and Computer Science*, 2011.

[8] Mohammad M. Masud, Qing Chen, Jing Gao, Latifur Khan, Jiawei Han, and Bhavani Thuraisingham "Classification and Novel Class Detection of Data Streams in a Dynamic Feature Space" in ISMIS 2009, LNAI 5722, pp. 552.

[9] Charu C. Aggarwal ,Jiawei Han, Jianyong Wang, Philip S. Yu "A Framework for On-Demand Classification of Evolving Data Streams" in ECML PKDD 2010, Part II, LNAI 6322, pp. 337–352.

[10] A. Azadeh , M. Saberi , A. Kazemc, V. Ebrahimipour , A. Nourmohammadzadeh, Z. Saberi "A flexible algorithm for fault diagnosis in a centrifugal pump with corrupted data and noise based on ANN and support vector machine with hyper-parameters optimization" Applied Soft Computing, Elsevier ltd. 2013. Pp 1478-1485.