ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

Data Stream Mining Big Data using Velocity Varying PSO Feature Selection

Shubham Khamitkar[1], Nikhil Badgujar[2], Vallabh Karanjkar[3], Hrishikesh Kherdekar[4]

B.E Student, Dept. of CSE, DYPIET, Pimpri, Pune, Savitribai Phule Pune University, Pune India[1234]

ABSTRACT: Big Data has a build-up up-springing numerous specialized difficulties that go up against both scholarly research groups and business IT sending, the abundant sources of Big Data are established on information streams and the scourge of dimensionality. It is for the most part realized that information which are sourced from information streams aggregate persistently making conventional cluster based model actuation calculations infeasible for continuous information mining. Highlight choice has been prominently used to ease the preparing burden in stir up an information mining model. On the other hand, regarding the matter of mining over high dimensional information the pursuit space from which an ideal element subset is inferred develops exponentially in size, prompting a defiant interest in computation. So, to handle this issue which is for the most part in view of the high-dimensionality and demonstrative arrangement of information bolsters in Big Data, a novel lightweight element determination is proposed. The component determination is composed especially to mine using so as to spill information on the fly, quickened molecule swarm advancement (APSO) sort of swarm pursuit that accomplishes improved diagnostic exactness inside sensible handling time. In this paper, an accumulation of Big Data with especially expansive level of dimensionality are put under test of our new component determination calculation for execution assessment.

KEYWORDS: Big Data, Particle Swarm Optimization, Swarm Intelligence, Feature Selection, Classification.

I. Introduction

Particle swarm optimization (PSO) is actually the population oriented heuristic search technique developed by Dr. Eberhart and Dr. Kennedy in 1995, motivated by actual behaviour of bird flocking. The PSO algorithm finds the global best solution by simply adjusting the physical phenomenon of each individual toward its own best location and toward the best particle of the entire swarm at every time. The PSO method had become very popular because of its simplicity in implementation as well as ability to quickly converge to an accordingly good solution. Since the PSO algorithm is easy to implement and efficient at the time of solving many optimization problems, it has attracted much attention. Many researchers have worked on improving its functionality in different ways, hence developing many interesting versions of PSO. The PSO method is becoming very popular because of its simplicity in implementation and ability to quickly converge to a reasonably good solution.

It has been observed that traditional PSO usually suffers from premature convergence i.e. tending to get stuck or blocked in local optima, low solution precision and so on. In order to avoid these issues and get better results, numerous improvements to PSO are proposed, that are mostly can be separated into two types. The first type, such as inertia weight, adaptive inertia weight and fuzzy inertia weight and so on, is to change the inertia weight (w) to make the algorithm that has strong global searching ability initially and also strong local searching ability in the end. The second type of improvement generally tries to change the structure of the algorithm or combine with other optimization algorithms (such as *genetic algorithm*) such as parallelizing PSO, Adaptive PSO and so on. So, those improved PSO always having better performance than the basic algorithm.

DOI: 10.15680/IJIRCCE.2016. 0405260

In case of continuous optimization, the variables in this model generally are nominally allowed to take on continuous range of values (usually real numbers). Continuous optimization problems are that problems which are typically solved using algorithms that generate a continuous sequence of values of the variables, called as iterates, that converges to a solution of an problem. In deciding the process of stepping from one iterate to the next, algorithm makes use of knowledge obtained at previous iterates(historical knowledge), and information related to the current model at the current iterate(ongoing), possibly including information about its sensitivity to disturbance or perturbation in the variables. This continuous type of the problem allows sensitivities to be defined in terms of first derivatives of the functions and second derivatives of the functions that define the models. In this paper a new method proposed using PSO with velocity to solve the optimization problem. Using the features of PSO and movements of accelerated particles, new method proposed and it gives optimal result.

II. Background

A. Motivation

Motivated from the natural behavior (like flocking of birds) swarm intelligence was born. Genetic Algorithm and Particle Swarm Optimization (PSO) were its two type. APSO is an extension to PSO were initialization process is accelerated to boost performance. PSO uses iterative approach which performs better than genetic algorithm for Single-mode Resource-Constrained Project Scheduling Problems

B. Big Data Datasets

It consists of three V’s: Velocity, Variety, and Volume. As the size of dataset increases the traditional algorithm not only fails in performance but also efficiency is lost. Big Datasets can be downloaded for UCI repositories. Using a labeled data set a classifier can be built. Half of the dataset are kept for training from which the classifier will be built while the other half is kept for testing which would evaluate our classifier for correctness and accurateness.

.

C. Traditional and Incremental Model Learning method

Traditional approach was top-down supervised learning. In this approach full set of data is used to construct classification model. The model is built based on stationary data set, any update in it requires repeating of whole process again. It performs well if datasets are stationary and no dynamic changes are anticipated to happen in near future. But in dynamic stream processing, data streams are evolving and thus the classification model would have to be frequently updated. So to solve this problem Data stream mining algorithm has been proposed.

In traditional method whole execution process is stored in runtime memory which would be problem if training data is too large. However, incremental method only load small fragment of input stream at time rather than taking full of it, so refreshing it is simple.

III.Proposed algorithm Incremental Learning Model

A. Batch-learning Classification Problem

Assume a data arrives for model induction from stream at time t, Dt is a vector of data with multiple attribute and class value $y_{t.}$

Thus, $D_t=[X^t,Y^t]$………………….(1)

Heuristic function is used for inducing a classification model. Let H(.) be a heuristic function. Here global optimal decision is attempted TRGLOBAL. Tree is global because whole data set is available.

Aim is to, *Maximize* $\Sigma^{M}_{i=1} \Sigma^{N}_{j=1} H(xij)$

where M: maximum no. of attribute,
N: maximum no. of instance received so far,
x_{ij}: splitting value,
and i <= M and j<=N
For newly arrived instance X_t at time stamp *t* the induced model will map it to predicated class y^t_k
Where k is set of all possible set of class.

TR_{GLOBAL} = Train (D_t,H(.))
y^t_k = Test(TR_{GLOBAL},X^t)
With every new data entry whole process needs to be repeated.

B. Incremental Learning Algorithm as Solution

Author in [12] had proposed an alternative method for incremental classification model induction, TR_{INCR}. Here only once training data is read without storing or loading it anymore. Here a tree is built by selecting an attribute for node splitting. This is done by computing Hoeffiding bound (HB) that checks that how often attribute value x_{ij} of attribute X_i would have corresponded to class y_k. It is also called as *Any-time algorithm.*

C. Popular Algorithm

Two main algorithm for incremental learning are: *functional-based* and *decision tree-based.*
Former group is likely to function as Black box and two most popular algorithms for *functional-based* are:

a. *KStar:*- Learns incrementally per instance.
b. *Updatable Naive Bayes*:- Based on assumption of possessing strong independence between the features. So it requires small amount of training data

Decision tree based algorithm uses *Any-time algorithm (HB)* discussed in last section.

D. APSO and Swarm Search

It is specially designed for choosing optimal subset from huge hyper-space. It is a wrapper-based feature-selection model and it retains accuracy by working on fitness value by picking higher fitness and deems the choice output. Our architecture is inspired from this approach. Basically we take a random feature subset in stochastic manner and flow enable to classification model and finally chosen feature subset converges. Fitness Evaluator is used which advises us how important candidate subset of feature is. If we use brute force method it will require large time to search every possible feature subset.
So to reduce time a new search base strategy named Swarm Search is used. Multiple agent work in parallel to speed up searching process.
Additionally speed-up is implemented in our model to speed up initialization step and thus names APSO.
In PSO there are two things:

a. Local best/individual best (o*i)
b. Global best (g*)

The reason for using individual best is to primarily increase diversity in quality solution**.**

IV. PSEUDO CODE

Support Vector Machines Pseudo Code

```
Initialize xI = P i∈I xi/|I| for every positive bag BI
REPEAT
Compute QP solution w, b for data set with positive examples {xI : YI = 1}
Compute outputs fi = hw, xii + b for all xi in positive bags
Set xI = xs(I), s(I) = arg maxi∈I fi for every I, YI = 1
WHILE (selector variables s(I) have changed)
OUTPUT (w, b)
```

Particle Swarm Optimization Pseudo Code

```
I) For each particle:
        Initialize particle and assign random space for each particle.
II) Do:
        a) For each particle:
                1. Calculate fitness value.
        2. If the fitness value is better than the best fitness value (pBest) in history
        3. Set current value as the new pBest
End
b) For each particle:
        1. Find in the particle neighbourhood, the particle with the best fitness
        2. Calculate particle velocity according to the velocity equation (1)
        3. Apply the velocity constriction
        4. Update particle position according to the position equation (2)
        5. Apply the position constriction
End
While maximum iterations or minimum error criteria is not attained.
```

V. Proposed Methodologies & Architecture

Our searching methodology was designed and implemented in the context of a Client Server system. We opted to store data in a MongoDB platform because it does not use the traditional table-based relational database structure but instead uses JSON-like documents because of their dynamic schemas (MongoDB calls that format BSON). This is used as it makes the integration of data in specific types of applications easier and also faster. This was particularly required as we were working on unstructured Big Data. We used various types of image formats for working along with plain text data and used SVM based classification in the PSO algorithm for usage on big data. We used the HTML, JavaScript and JSP for the designing purposes of the Graphical User Interface.

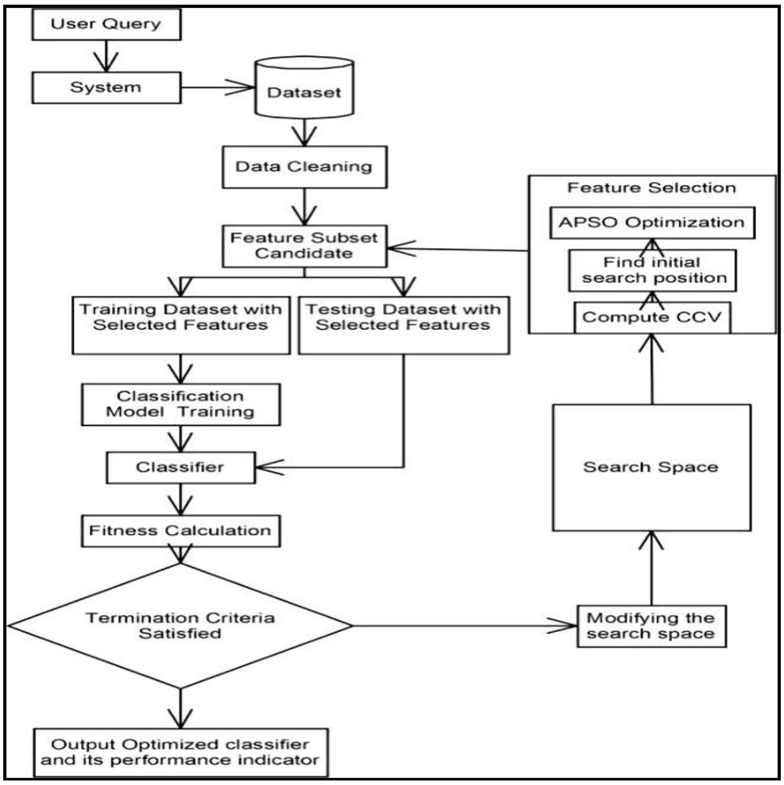

User Query
System
Dataset
Data Cleaning
Feature Subset Candidate
Feature Selection
APSO Optimization
Find initial search position
Compute CCV
Training Dataset with Selected Features
Testing Dataset with Selected Features
Classification Model Training
Classifier
Search Space
Fitness Calculation
Termination Criteria Satisfied
Modifying the search space
Output Optimized classifier and its performance indicator

Fig. 1. System Architecture

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

VI. Results

The results showed that the proposed algorithm performs better as compared to existing system in searching the data from unstructured dataset.In our proposed system, end-user will get the result of entered query in the form of structured data as an optimised classifier which is derived by influence of Particle Swarm Optimization (PSO) technique.If we compare the existing system with our proposed system, it is found that efficiency of the proposed system has increased to a considerable extent. If we compare robust hash method to our proposed system, then the efficiency increases from 80.1 % to 87.9%.

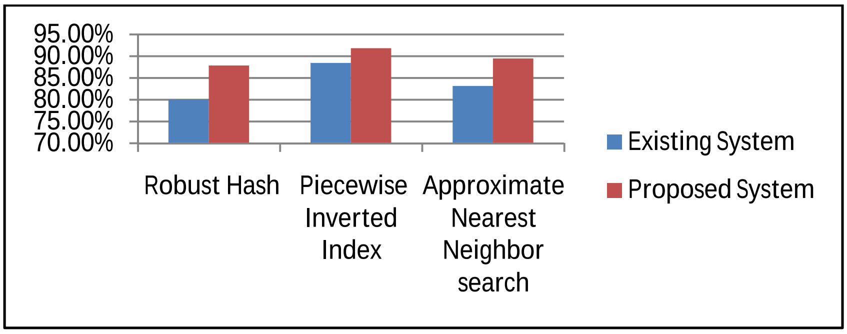
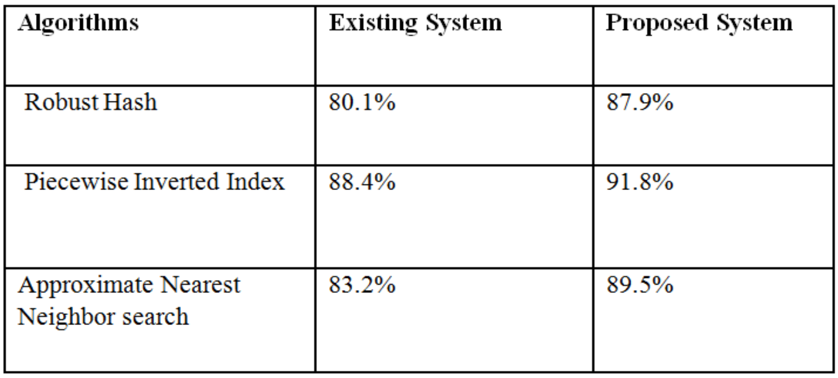


Fig. 2. Efficiency of proposed system against various existing systems.



Algorithms	Existing System	Proposed System
Robust Hash	80.1%	87.9%
Piecewise Inverted Index	88.4%	91.8%
Approximate Nearest Neighbor search	83.2%	89.5%

Fig. 3. Efficiency of POS tagging against Hash tagging.

VII. Conclusion and Future Work

In Big Data investigation, the high dimensionality and the spilling way of the approaching information disturb awesome computational difficulties in information mining. Enormous Data becomes persistently with crisp information are being produced at all times; henceforth it requires an incremental calculation approach which has the capacity screen expansive size of information powerfully. Lightweight incremental calculations ought to be viewed as that is equipped for accomplishing vigor, high exactness and least pre-processing inactivity. In this paper, we explored the likelihood of utilizing a gathering of incremental grouping calculation for characterizing the gathered information streams relating to Big Data. As a contextual investigation experimental information streams were spoken to by five datasets of distinctive do-primary that have expansive measure of components, from UCI file. We analysed the conventional grouping model prompting and their partner in incremental actuations. Specifically we proposed a novel lightweight element choice system by utilizing Swarm Search and Accelerated PSO, which should be valuable for information stream mining.

Right now we have used APSO for text classification as well as for a image (as a feature) classification. For the future scope we can add variety of data such as structure, semi-structure, un-structure data for optimization. We are planning to extend our project to optimize video processing, audio processing, as well as variety of data that are not currently supported in this model.

DOI: 10.15680/IJIRCCE.2016. 0405260

REFERENCES

1. Simon Fong, Raymond Wong, and Athanasios V. Vasilakos "Accelerated PSO Swarm Search Feature Selection for Data Stream Mining Big Data",DOI 10.1109/TSC.2015.2439695,IEEE Transactions on Services Computing, 2015.
2. J. Ross Quinlan.R., C4.5: "Programs for Machine Learning."Morgan Kauf-mann Publishers, pp. 1-6, Inc., 1993.
3. Mohamed Medhat Gaber, Arkady Zaslavsky, Shonali Krishnaswamy, "Mining data streams: a review", ACM SIGMOD Record, Volume 34 Issue 2, pp.18-21,2005.
4. Wei Fan, Albert Bifet, "Mining Big Data: Current Status, and Forecast to the Future", SIGKDD Explorations, Volume 14, Issue 2, pp.1-5, December2012.
5. John G. Cleary, Leonard E. Trigg: "K*: An Instance-based Learner Using an Entropic Distance Measure."In: 12th International Conference on Machine Learning, pp.108-114, 1995.
6. Bifet A. and Gavalda R. "Learning from time-changing data with adaptive windowing". In Proc. of SIAMInternational Conference on Data Mining, Volume-7, pp. 443-448, 2007.
7. Indre Zliobaite, Albert Bifet, Bernhard Pfahringer, Geoff Holmes, "Active Learning with Evolving Streaming Data", Volume 6913, pp.597-612, 2011.
8. Simon Fong, Suash Deb, Xin-She Yang, Jinyan Li, "Metaheuristics Swarm Search for Feature Selection in Life Science Classification", IEEE IT Professional Magazine, Volume 16, Issue 4, pp.24-29, August 2014.
9. Xin-She Yang, Suash Deb, Simon Fong, "Accelerated Particle Swarm Optimization and Support Vector Machine for Business Optimization and Applications", The Third International Conference on Networked Digital Technologies (NDT 2011), Springer CCIS 136, Macau, China, Volume 136, pp.53-66, July 2011.
10. Fong, S., Liang, J., Wong, R., Ghanavati, M., "A novel feature selection by clustering coefficients of variations",*Ninth International Conference on Digital Information Management (ICDIM)*,DOI, 10.1109/ICDIM.2014.6991429, pp.205-213, Sept. 2014.
11. I.H. Witten, E. Frank, "Data mining: practical machine learning tools and techniques with Java implementations", Morgan Kaufmann, J.S. Bridle, "Probabilistic Interpretation of Feed forward Classification Network Outputs, with Relationships to Statistical Pattern Recognition", 2005,F. Fogel-man-Soulie and J. Herault, "*Neuro-computing—Algorithms, Architectures and Applications* eds., NATO ASI Series F68, Berlin: Springer-Verlag, (Book style with paper title and editor) pp. 227-236, 1989.
12. Domingos P., and Hulten G. 2000. "Mining high-speed data streams", in Proc. of 6th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'00), DOI.10.1145/347090.347107, pp. 71- 80, 2000.