Batch -Incremental Classification of Stream Data Using Storage

Parita Ponkiya† and Rohit Srivastava††,

GTU, PIET, Vdodara, India

Abstract:
Data mining is a technique that is used to extract useful knowledge from large amount of data. And classification is most important task of data mining. Now a day's in real world stream data is most important source of knowledge. Stream data is data that continuously arrives over the time i.e. growth of data is increasing faster and faster. Traditional classification algorithms are not suitable for such data. Continuous growth of the data makes previously constructed classification tree outdated and it is to be reconstructed from the scratch, which is very time consuming. Another major issue is the data-type, as each of them is to be treated separately, among which the continuous data produces major challenge in the tree building task, needs to be discretized. Out of many classifications algorithms, ID3 is a famous tree based classification algorithm which deals with only categorical data and uses information gain for attribute selection. In this paper the tree based Batch incremental classification algorithm is proposed for stream data that outputs tree same as ID3. It uses CAIM based discretization for continuous attributes and various attribute selection criterions along with storage structure for the strategic information of every node and the historical data to rebuild decision tree. CAIR, CAIU, CAIM criterions are used as attribute selection criterions and comparison is also provided between these attribute selection measures.

Key Words:
Classification, CAIR, CAIM, CAIU, Information Gain, Batch Incremental Classification

1. INTRODUCTION

Data mining has been identical as the technology [4] that offers the possibilities of discovering the hidden knowledge from this large amount of data. Four major task of data mining are classification, clustering, regression and association. Classification is important task in data mining, there are various classification techniques are available like decision tree induction, Bayesian networks, k-nearest neighbor classifier and fuzzy logic techniques. Among these models decision tree is suitable model because it is simple and easy to understand for human beings, and can be constructed really fast. But there are many issues while building a tree from stream data.

Stream data [3] are data streams which is generates data speedily, that data is growing up faster and faster. It encounters a continuous stream of real-time data from multiple sources. Stream data is an important source of knowledge that enables us to take important decision in real time. [2]

Traditional tree based classification [7] problem consist of building a decision tree from already available instances i.e. from static dataset, so it is assumed that dataset contains all the information at the time of building classification model. Over the time change may occur in concept of instances, which model is trying predicting, sudden change characteristics of data is called concept drift, so this makes previously classified decision tree model inaccurate.

This problem can be solved by building a decision tree classification model incrementally. Incremental decision tree allows updating an existing tree without building a tree from scratch. Instead of building a decision tree from static data, building a decision tree from the data that is available over time. Tree can be built incrementally in two ways either instance by instance or in batches. This type of learning is called incremental learning. Incremental decision tree learning May generates decision tree model accurate as it generates from stream data and also able to detect concept drift.

2. RELATED WORK

Extensive research is done for various classification and incremental classification algorithms. One of the main drawbacks with the classical tree induction algorithms is that they do not consider the time in which the data arrived. This draw back motivated researcher to develop method which update decision tree classification model as new data arrives instead of rerunning algorithm from scratch that results incremental classification. [7]

ID3 [11] is most popular classification algorithm. Various incremental classification algorithms such as ID4 [11], ID5 [11], ID5R [11], ITI [10], Vfdt [7], Cvfdt [7] are proposed by reserchers. An incremental classification can be defined as ID3 compatible if it build almost similar decision tree as produced by ID3 using all training set. This approach is maintained by classification method such as ID4 [11], ID5 [11] and ID5R [11].

Manuscript received April 5, 2015
Manuscript revised April 20, 2015

There are many discretization algorithms are available, they are mainly categorize into supervised and unsupervised algorithm. CAIM discritiztion algorithm is batter compared to other algorithm [8]

3. DISCRETIZATION PROCESS

Discretization [8] is the process of splitting numeric values into discrete intervals with non-overlapping interval boundaries. The intervals themselves are treated as ordered and discrete values. You can discretize both numeric and string columns. And this discretization process can be used as front end of machine learning algorithm and in back end decision tree can be constructed using one of the algorithms like ID3, C4.5 etc. [4]

The CAIM algorithm aims to:

- Maximize the interdependency between the continuous valued attribute and its class labels,
- Achieve the minimum number of discrete intervals possible, and Perform the discretization task at reasonable computational cost so that it can be applied to continuous attributes with large number of unique values.

Equation for CAIM is defined as:

$$CAIM(C, D|F) = \frac{\sum_{r=1}^{n} \frac{max_r^2}{M_{+r}}}{n}$$

The Class-Attribute Interdependence Redundancy (CAIR) measure is defined as [8]

$$CAIR(C, D|F) = I(C, D|F) /H(C, D|F)$$

The Class-Attribute Interdependence Uncertainty (CAIU) measure is defined as [8]

$$CAIU(C, D|F) = INFO(C, D|F) /H(C, D|F)$$

4. PROPOSED BATCH INCREMENTAL CLASSIFICATION ALGORITHM

The purpose of this paper is to present modified Tree Based ID3 algorithm, which uses different attribute selection criterions such as CAIM, CAIR, CAIU and Information gain to build tree incrementally instead of only information gain used in ID3 algorithm.

Algorithm:

```
1. Read Input Data
2. If attribute is continuous, discretize it using CAIM
3. Calculate CAIR/CAIM/CAIU value for that attribute
4. Let A be the attribute with the highest information gain or caim/cair/caiu
5. Create a decision node that splits on A
6. Recurse on the sublists obtained by splitting on A, and add those nodes as children of node
7. For each node in tree
        Insert tree information in to database
        Insert attribute, cair value of attribute, qmatrix of every attribute into database
8. Retrieve the stored model tree from database
9. For each attribute a in recently arrived block of data
10. Read the block of continuous data
11. Discritize continuous attribute
12. Apply a block to model tree
13. Update tree, cair and qmatrix information in database
14. If drift is detected then
        Add branch or restructure the tree
15. Output the classification result
```

In this proposed algorithm, if attribute is continuous then it discritize attribute as shown in previous section.

Then it selects highest value of CAIR/CAIM/CAIU. Then recursively apply this algorithm on each block of data to build decision tree incrementally.

Figure1 shows the general proposed frame work for incremental classification process in which it reads the input data and based on that data it trains classification model, after that incremental classification model is trained based on classification model which will read stream data to train incremental classification model.

Figure 2 shows train classification model which reads the data, if data contains continuous attribute then it will descretize continuous attribute and also stores attribute information into database. After that tree is build from discrete attribute value and model information, cair value information and qmatrix value information is stored into database, and finally it outputs the classification result.

Figure 3 shows incremental classification model which first retrieves the stored model tree then it reads the block of continuous data and then it will descretize continuous attribute after that it will apply block of data to model tree and updates all the information about model into database. If drift is detected then it will add branch or restructure the tree and outputs the classification result.

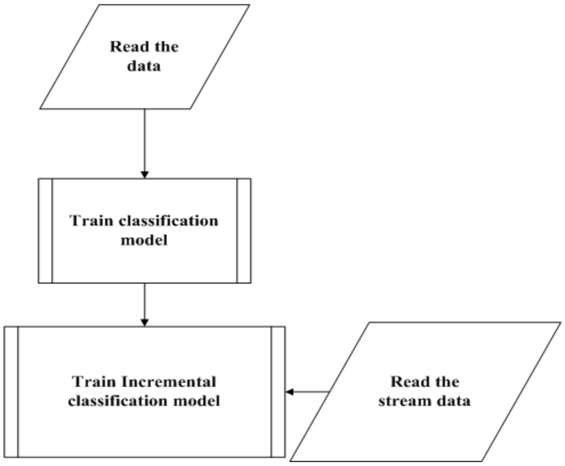
Read the data
Train classification model
Train Incremental classification model
Read the stream data

Figure1. Proposed framework for incremental classification

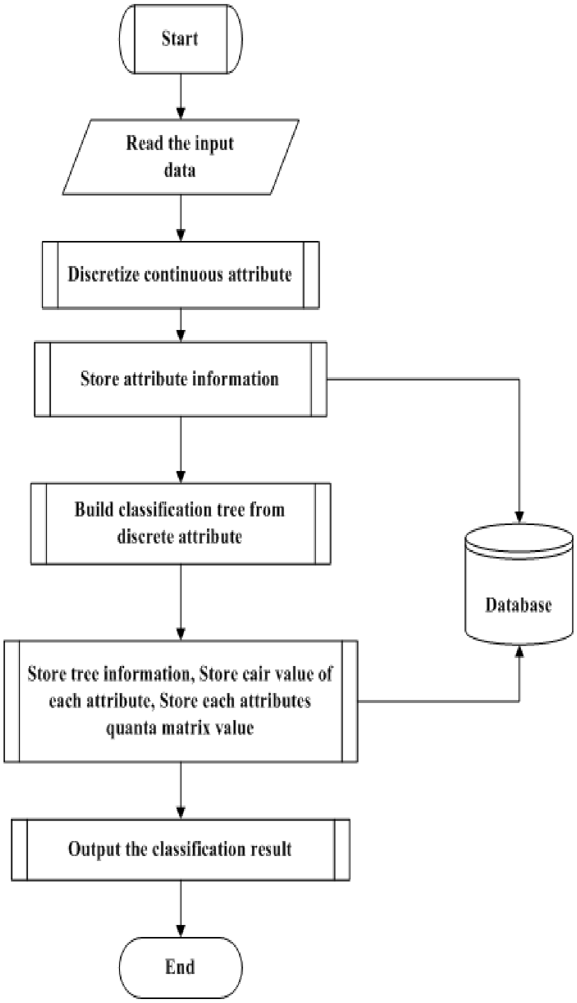
Start
Read the input data
Discretize continuous attribute
Store attribute information
Build classification tree from discrete attribute
Database
Store tree information, Store cair value of each attribute, Store each attributes quanta matrix value
Output the classification result
End

Figure2. Train classification model

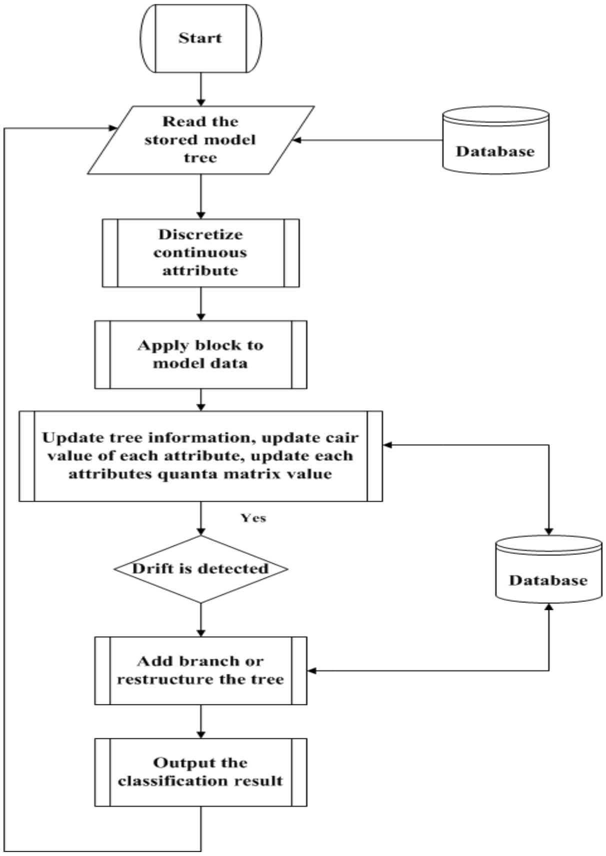
Start
Read the stored model tree
Database
Discretize continuous attribute
Apply block to model data
Update tree information, update cair value of each attribute, update each attributes quanta matrix value
Yes
Drift is detected
Database
Add branch or restructure the tree
Output the classification result

Figure3. Incremental classification model

5. RESULT

In this section, the result of proposed algorithm with various attributes selection criterions such as CAIM, CAIR, CAIU and information gain on some datasets with some continuous attributes are presented. Accuracy and execution time comparison for block incremental classification is done. Accuracy and execution time comparison for different attribute selection measure is presented

Experimental setup

To perform the proposed method, java is used in MyEclipse 7.5. MyEclipse 7.5 [12] is used as front end and oracle 11g r2 [13] is used as back end to provide efficient storage structure. For the preprocessing task like missing value replacement weka tool is used, before applying to these algorithms. Table 1 shows the description of dataset used in experiments

Table 1 Dataset Description

Dataset	No. of attribute	No. of continues Attribute	No. of Records
Ozone	73	72	2534
Adult	15	3	32561
KDD	32	32	100000

After preprocessing dataset using weka, training dataset is applied to the implemented tool. First it builds classification tree and after that block of data is applied to classified model incrementally which updates the tree and outputs the classified result. Tool shows classification tree, correctly incorrectly and not classified instances, accuracy and execution time.

Result Analysis

Table 2 Accuracy measure with window size 200, 400, 600, 800 of ozone dataset

Dataset	Block size	IG	Cair	Caim	Caiu
ozone	200	91.92	91.79	92.46	93.13
	400	94	89.9	93	95.15
	600	91.11	95.67	95.06	96.11
	800	88	94.94	93.44	93.25

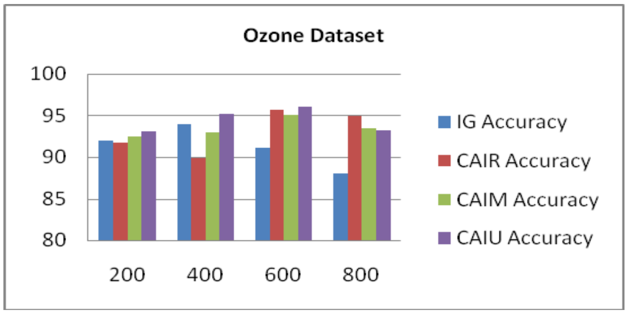

Figure4. Ozone dataset accuracy comparison with different window size

Table 3 Accuracy measure with window size 200, 400, 600, 800 of Adult dataset

Dataset	Block size	IG	Cair	Caim	Caiu
Adult	200	65.02	71.62	60.66	64.11
	400	67.7	70.92	60.7	70.35
	600	62.97	67.92	59.94	66.71
	800	88	94.94	93.44	93.25

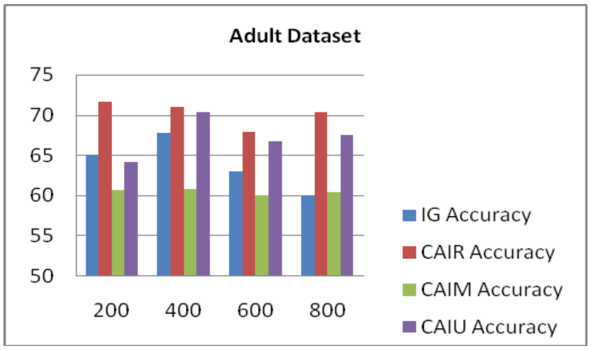

Figure5. Adult dataset accuracy comparison with different window size

Table 4 Accuracy measure with window size 200, 400, 600, 800 of KDD dataset

Dataset	Block size	IG	Cair	Caim	Caiu
KDD	200	89.2	80.09	85.05	89.53
	400	90.12	96.07	96.06	90.52
	600	60.71	84.66	66.77	54.51
	800	82.03	89.38	67.56	84.65

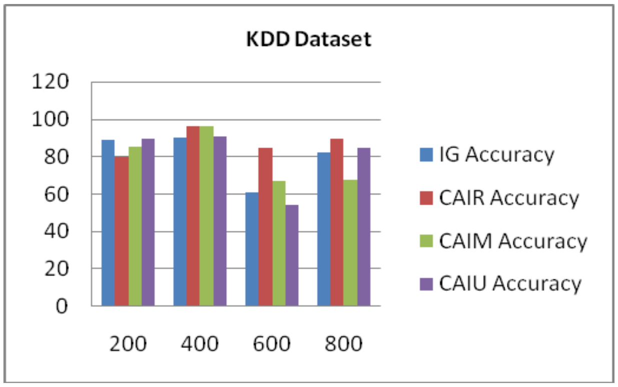

Figure6. KDD dataset accuracy comparison with different window size

Table 5 Execution time (ms) measure with window size 200, 400, 600, 800 of Adult dataset

Dataset	Block size	IG	Cair	Caim	Caiu
Adult	200	387.83	437.04	624.72	534.01
	400	1293.8	1286.56	1607.8	1520.95
	600	2373.5	2283.88	2503.50	3126.75
	800	4982.9	4699.71	4010.38	5778.46

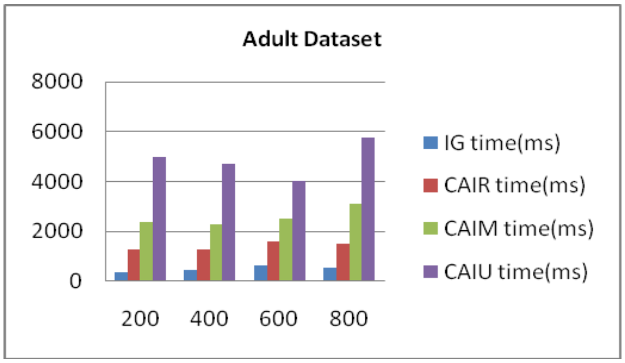

Figure7. Adult dataset execution time comparison with different window size

From the accuracy comparison results, it is easily detected that cair gives more promisible result than all other attribute selection method.

Execution time is also compared of other data sets. Information gain execution time is less compared to other attribute selection methods because it's not memory residential while other three methods are memory residential.

6. CONCLUSION

In this paper the Tree based Batch Incremental classification algorithm for stream data is proposed which use CAIM as the discretization process and it is tested with different attribute selection criterions such as CAIR, CAIU, CAIM and Information Gain for the classification accuracy. The model is incrementally updated for the blocks of the data stream. The efficient storage structure for the classification model tree allows model updation and test for the drift of trend in data. The restructured model for the latest data will take place of the actual classification tree for the forth coming data. Classification accuracy is tested for different datasets with different block size for all the model trees created. The results are analyzed and CAIR is found to be more suitable for classification accuracy for continuous and categorical data as well.

REFERENCES

[1] T. Ryan Hoens · Robi Polikar · Nitesh V. Chawla , " Learning from streaming data with concept drift and imbalance: an overview" , Springer, 13 January 2012

[2] Mohamed Medhat Gaber, " Advance in data stream mining ", Springer, 2012

[3] Joao Gama, " A Survey on learning from data streams: Current and future treands " ,Springer,2012

[4] M.R Lad, R.G Mehta, D.P Rana, "Novel Tree Based Classification", IJESAT, 2012.

[5] Ryan Elwell, Robi Polikar, "Incremental Learning of Concept Drift in Nonstationary Environments", IEEE VOL. 22, NO. 10, OCTOBER 2011

[6] Sheng Chen • Haibo He, " Towards incremental learning of nonstationary imbalanced data stream: a multiple selectively recursive approach" , Springer 2010

[7] Ahmed Sultan Al-Hegami , " Classical and incremental classification in data mining process ", IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.12, December 2007

[8] L. Kurgan and K.J. Cios, "CAlM Discretization Algorithm", IEEE Transactions of Knowledge and Data Engineering, Vo1.16, No.2, February 2004.

[9] Hankil Yoon, Khaled Alsabti, Sanjay Ranka, "Tree based classification for large datasets",1999

[10] P. E. Utgoff, "An improved Algorithm for Incremental Induction of Decision Tress",. Springer 1994.

[11] P. E. Utgoff, "Incremental Induction of Decision Tress", Machine learning, 4(2), pp 161-186. Springer 1989.

[12] Official site of http://www.myeclipseide.com/

[13] http://www.oracle.com/technetwork/database/database-technologies/express-edition/overview/index.html