Stream-based Biomedical Classification Algorithms for Analyzing Biosignals

Simon Fong*, Yang Hang*, Sabah Mohammed** and Jinan Fiaidhi**

Abstract—Classification in biomedical applications is an important task that predicts or classifies an outcome based on a given set of input variables such as diagnostic tests or the symptoms of a patient. Traditionally the classification algorithms would have to digest a stationary set of historical data in order to train up a decision-tree model and the learned model could then be used for testing new samples. However, a new breed of classification called stream-based classification can handle continuous data streams, which are ever evolving, unbound, and unstructured, for instance--biosignal live feeds. These emerging algorithms can potentially be used for real-time classification over biosignal data streams like EEG and ECG, etc. This paper presents a pioneer effort that studies the feasibility of classification algorithms for analyzing biosignals in the forms of infinite data streams. First, a performance comparison is made between traditional and stream-based classification. The results show that accuracy declines intermittently for traditional classification due to the requirement of model re-learning as new data arrives. Second, we show by a simulation that biosignal data streams can be processed with a satisfactory level of performance in terms of accuracy, memory requirement, and speed, by using a collection of stream-mining algorithms called Optimized Very Fast Decision Trees. The algorithms can effectively serve as a corner-stone technology for real-time classification in future biomedical applications.

Keywords—Data Stream Mining, VFDT, OVFDT, C4.5 and Biomedical Domain

1. Introduction

Emerging biomedical applications (e.g. real-time tele-health-monitoring systems, advanced clinical decision-support systems, and mobile biosignal analyzers) demand for an efficient classification model that is capable of interpreting the data streams and classifying unseen samples into pre-defined groups. What these applications have in common is the need to process infinite biosignal data streams and to extract insights out from them in real-time. Most of the research papers from the literature of bioinformatics are based on traditional classification models that may run short of meeting the real-time requirements and supporting the stream processing features [1].

A traditional classification model refers to the method of top-down supervised learning where the full set of stationary data is used to construct the relations between the attributes and the

Manuscript received November 17, 2011; accepted November 22, 2011.

Corresponding Author: Simon Fong

* Department of Computer and Information Science, University of Macau, Macau SAR (ccfong@umac.mo)

** Department of Computer Science, Lakehead University, Thunder Bay, Canada (sabah.mohammed@lakeheadu. ca)

classes into a decision tree, by recursively partitioning the dataset into conditional nodes and paths. Because the decision tree is built based on a stationary set of data, updating the tree needs to repeat the whole training process when the dataset is added with new data in order to incorporate the changing underlying patterns that are hidden in the new data. The traditional models might have worked well for supporting decision-support systems that are largely grounded on a huge volume of historical data, and where the data is relatively stationary. However, in a dynamic stream processing environment, data streams are ever evolving and the decision tree would have to be frequently updated accordingly. Therefore a new generation of algorithms, generally known as stream-based classification algorithms, have been proposed to solve this problem. The essence of stream-based classification is its unique tree building process that starts from scratch and it grows incrementally to a full tree while the data progressively streams in. Each segment of the data from the stream is read one pass at a time and updating the tree structure does not require accessing the seen data again.

In this paper we evaluated a number of classification algorithms that are both case-based and traditional, in the light of analyzing biosignal data streams. An improved version of stream-based classification called Optimized Very Fast Decision Tree (OVFDT) from the original VFDT was tested for the first time for handling biosignal data streams. OVFDT has already demonstrated its superiority over other versions of VFDT, and it is anticipated that it may do equally well with biosignal data streams. The paper contributes to biomedical research communities as a pioneer work, as well as serving a guideline for other researchers who may be interested in employing stream-based classification algorithms for developing future real-time biomedical applications.

The remainder of this paper is organized as follows: related work on biomedical classification is presented in the next section. A general framework for stream mining biosignals is described in Section 3, as are the corresponding stream-based classification algorithms. In Section 4 we present some experimental results of the comparison between the stream-based classification algorithms, and finally Section 5 concludes this paper.

2. Related Work

An uprising research trend that was recently observed is designing mobile biomedical devices. Two examples are [2] and [3], which focus on a handheld mobile electrocardiogram (ECG) analysis device and remote patient monitoring with mobile real-time clinical decision support respectively. They need to have a core function that can classify or predict a result interactively by receiving and mining dynamic data streams, while the "real-time" requirements are being emphasized. Unfortunately, most, if not all, of the past works were assuming traditional classification algorithms. Thus, it is doubtful whether the existing classification algorithms would be able to catch up with the fast rising trend that demands for real-time responses and stream processing. It is equally curious to know too, if the alternative solution offered by stream-based classification is as effective as anticipated.

Zwaag van der et al. had compared a number of past papers that had contributed to biomedical classification in his recent guideline on biosignal driven HCI [4] that was published in 2010. The findings show that in 2008 and 2009 researchers were attempting to data mine biosignals including cardiovascular activity, electrodermal activity, respiration, electromyograms, and skin temperature for classifying different affective emotions. The classification techniques used as reported in [4] are mainly the Support Vector Machine and Linear Discriminant Analysis. Just like traditional decision trees, these classification techniques require iterative computation over the whole dataset in order to produce a result. However, the accuracies range from only 47% to 83% by those techniques, because such techniques have limitations of scaling up to high dimensionalities for representing non-linear relations that may exist in the data. Lee et al., in 2007 [5], attempted to data mine an ECG biosignal for 3 positions, by using a combination of feature selection techniques (for reducing the dimensionalities and retaining the relevant attributes) and the C4.5 decision tree. The classifier was able to do coronary artery disease diagnosis at an accuracy of 90%, which was higher than the predecessors. In the same year, Chen et al. built a biomedical classifier by integrating it with association rules and a sequential pattern [6]. The classification model received real-time environmental or atmospheric data and it output three levels of warming for asthma prediction. The peak accuracy for it is 87.52%. It is speculated that these classification examples come short of embracing the latest data into the training model as the model was trained by old data some time ago. So far to the best of the authors' knowledge no attempt has been made in using stream-based classification algorithms in biomedical analysis.

3. Proposed Framework for Stream-based Classification

A high level view of the workflow of stream-based classification is introduced here. Stream-based classification refers to one that can meet the challenges of processing unbounded, infinite, real-time biosignals. The requirement for acquiring timely decisions from the classifier is that the mining time must be shorter than the rate of the incoming data streams. The other requirement is that we do not assume that a full set of data is always available. Hence this type of stream-based algorithm fits the bill as it can process one pass of data at a time, and a decision is generated instantly with certain accuracy. These requirements are typical in biomedical applications especially those that involve real-time monitoring and instant analysis. A typical workflow is depicted in Fig. 1.

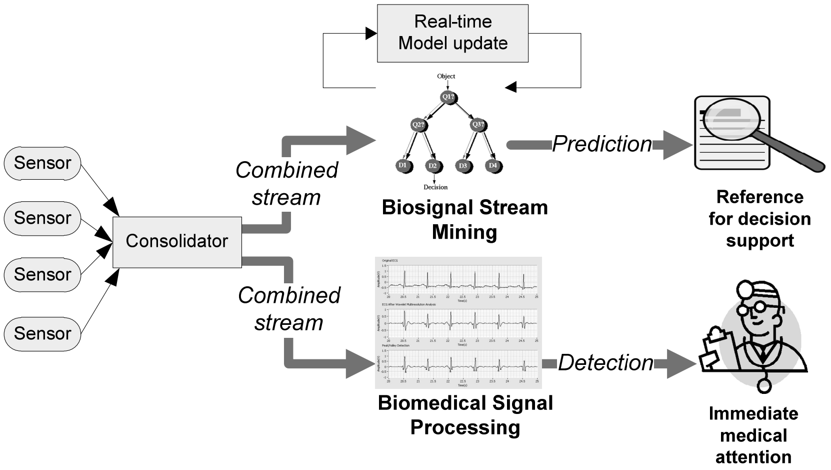


Fig. 1. Workflow of a real-time biomedical application that utilizes stream-based classification

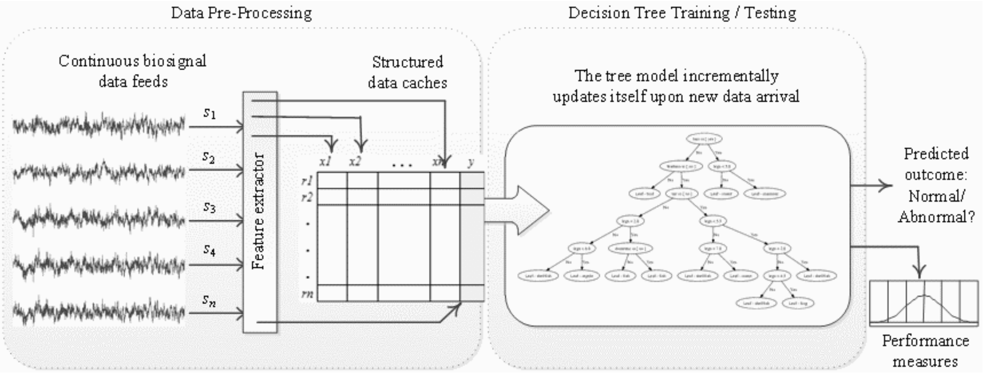

Data Pre-Processing
Decision Tree Training / Testing
Continuous biosignal data feeds
Structured data caches
s_1
s_2
s_3
s_4
s_n
Feature extractor
x1
x2
...
xn
y
r1
r2
rn
The tree model incrementally updates itself upon new data arrival
Predicted outcome: Normal/ Abnormal?
Performance measures

Fig. 2. Data Stream Mining for biosignal stream classification

While fresh data collected by the sensors streams in, the classification model updates itself simultaneously. The data is read in one pass each time, in a manner of read-train-and-forget. No database is needed for storing the historical data. The stream mining model can be optionally extended to couple with some signal processing and detection algorithm, for picking out anomalous patterns from the incoming streams. The output of the classification can be used as references or as advice in decision support. One advantage of this workflow is that it can operate in real-time that encompassed feeding in the data streams, processing them, and using the data to train/refresh the decision tree so that both detection and classification can be conducted simultaneously.

The next diagram in Fig. 2 describes a lower level operation of biosignal stream mining. It essentially indicates that the incoming data streams would have to be synchronized and temporarily cached in a buffer with the attributes and their values appropriately identified. The data in the cache are queued to enter the classifier one segment at a time. Once the data reaches the classifier the data samples will traverse the decision tree from the root to the bottom. The statistics accumulated at each node would be updated and if the conditions are sufficiently met the leave node at the bottom will turn into a decision node - this is called node splitting. Therefore both classification (prediction) and a decision tree model update (refresh) will happen simultaneously when a segment of data samples pass through the tree.

3.1 Formulation of the Stream-based Decision Tree

The decision tree in stream-based classification is built increasingly from the incoming data stream a small amount at a time, by splitting up a node into two. How many samples that have been seen by the learning model in order to expand a node, depend on a statistical method called Hoeffding bound or additive Chernoff bound. This bound is for deciding how many samples are statistically required at each node for a split. As the data arrives the tree is evaluated and its tree nodes could be expanded on the fly. The following equations essentially depict the building blocks of the stream mining model, which uses the Hoeffding bound. The Heuristic evaluation function is used to judge when a leaf at the bottom of the tree is converted to a conditional node, thereby it then pushes up the tree. A node split occurs when there is sufficient evidence that a new conditional node is needed, so replacing the terminal leaf by the relevant decision node better reflects the current conditions as represented by the rules in the tree.

Let *G(.)* denote the heuristic evaluation function for building a decision tree based on the Information Gain of an attribute, $I(A_j)$. The $I(A_j)$ function measures the amount of information that is sufficient to classify a sample as a node by the theory of information gain. The merit of a discrete attribute's counts n_{ijk}, representing the number of samples of class k that reach the leaf where the attribute j takes the value i which is estimated by sufficient statistics. In Equation (3), P_i is the probabilities of observing the value of attribute i. P_{ik} is the probability of observing the value of the attribute i given class k.

$$G(A_j) = I(samples) - I(A_j) \tag{1}$$

$$I(A_j) = \sum_i P_i (\sum_k - P_{i,k} \log(P_{i,k})) \tag{2}$$

$$P_{i,k} = n_{i,j,k} / \sum_a n_{a,j,k} \tag{3}$$

$$P_i = \sum_a n_{i,j,a} / \sum_a \sum_b n_{a,j,b} \tag{4}$$

$$\varepsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2N}} \tag{5}$$

Let us assume that we have a real-valued random variable r with a bounded range of R, which arrives at the n number of independent observations. Equation (5) shows how the Hoeffding bound is computed, with a confidence level of $1\text{-}\delta$, and a mean of r is $r\text{-}\varepsilon$ at least. The observed mean of samples is r'. We assume that the range R has a probability of 1 given that the information gain of R is $\log_2$Class#. The core of the algorithm is the use of the Hoeffding bound for choosing a split attribute as the decision node. We let x_a be the attribute that has the highest $G(.)$, and x_b be the attribute that has the second-highest $G(.)$. Such that the difference between the pair of the top quality attributes is defined as $\triangle G = G(x_a) - G(x_b)$.

3.2 Formulation of OVFDT and Functional Tree Leaves

OVFDT is extended from [7], which is an optimized version of VFDT known as the Optimized Very Fast Decision Tree. The formulation is exactly the same as VFDT except a few modifications as described below. Essentially, it can produce a balance of good accuracy and compact tree size with these modifications.

Modification 1. An adaptive tie-breaking threshold is used for node-splitting control by incremental computing. It modifies the attribute-splitting process by using a dynamic tie-breaking threshold instead of a user-defined fixed value. The dynamic tie is computed as the mean of the Hoeffding bound, which is obtained by a new node splitting corresponding to a certain class. Previous experiments have shown that the OVFDT tree learns as fast as the original VFDT.

Modification 2. An incremental pruning mechanism is used to solve the explosion of the tree size. The splitting is constrained by a lightweight pre-pruning mechanism, because post-pruning is inappropriate for the non-stopping data streams operation.

Modification 3. OVFDT can apply Functional Leaf strategies [8] in the prediction phase to test a sample like how the VFDT does. A classification problem is defined as follows: *N* is a set of examples in the form (*X*, *y*), where *X* is a vector of *d* attributes and *y* is a discrete class label. The classification goal is to produce a decision tree model from *N* examples, which predicts the classes of *y* for future examples and *x* for high accuracy. In data stream mining, the example size is very large or unlimited, N→∞. Suppose n_{ijk} is the sufficient statistics of attribute X_{ij} to y_k. When a new instance arrives, it is sorted to leaf *l* using the current Hoffding Tree (HT). If the sorted class is the same as the actual class label in the instance, it means the instance is truly predicted by HT; otherwise, it is falsely predicted.

As shown in Fig. 3, the node splitting mechanism is a feature in OVFDT and the additional prediction strategy that is installed as the Functional Leaf is installed at different steps of the operational flow. OVFDT is concerned about the controlling node splitting in the tree induction phase, while the Functional Leaf independently functions at the prediction/testing phase. In addition to the *Majority Class* and *naïve Bayes* strategies, an adaptive *Hybrid* strategy is also proposed. The *Hybrid* strategy is able to shift between a *Majority Class* and *naïve Bayes* strategies, which are described in Fig. 4.

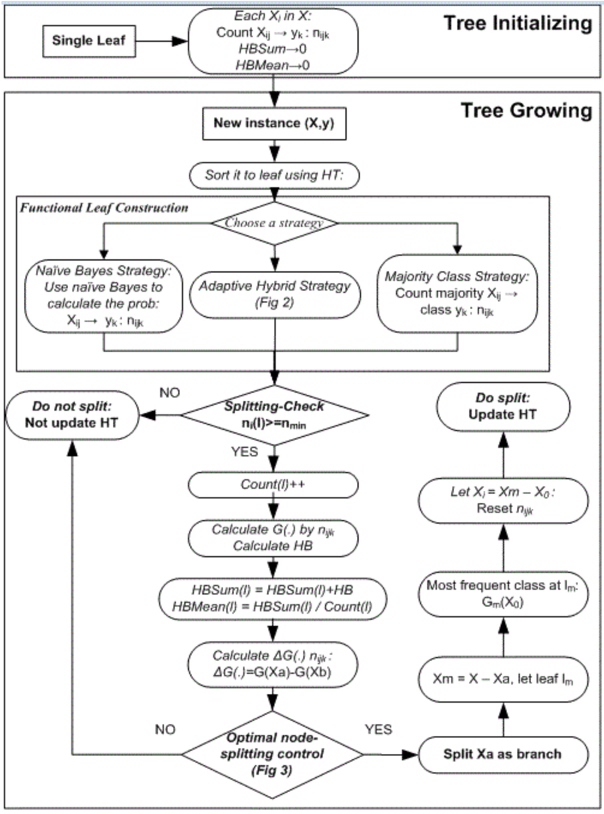

Tree Initializing
Single Leaf
Each X_i in X:
Count X_ij → y_k : n_ijk
HBSum→0
HBMean→0
Tree Growing
New instance (X,y)
Sort it to leaf using HT:
Functional Leaf Construction
Choose a strategy
Naïve Bayes Strategy: Use naïve Bayes to calculate the prob: X_ij → y_k : n_ijk
Adaptive Hybrid Strategy (Fig 2)
Majority Class Strategy: Count majority X_ij → class y_k : n_ijk
NO
Do not split: Not update HT
Splitting-Check n_i(l)>=n_min
YES
Count(l)++
Calculate G(.) by n_ijk
Calculate HB
HBSum(l) = HBSum(l)+HB
HBMean(l) = HBSum(l) / Count(l)
Calculate ΔG(.) n_ijk :
ΔG(.)=G(Xa)-G(Xb)
NO
YES
Optimal node-splitting control (Fig 3)
Split Xa as branch
Xm = X – Xa, let leaf l_m
Most frequent class at l_m: G_m(X_0)
Let X_i = Xm – X_0 : Reset n_ijk
Do split: Update HT

Fig. 3. Flow diagram of OVFDT with Functional Tree Leaf Learning

The optimal node-splitting control consists of an adaptive tie and a pre-pruning mechanism. During each node-splitting, the Hoeffding bound (HB) value that relates to leaf *l* is recorded. In OVFDT, the recorded HB values are used to compute the adaptive tie, which uses the mean of HB to each leaf *l,* instead of a fixed user-defined value in VFDT. Suppose T_i is the true prediction of instance in the i^{th} splitting estimation, and F_i is the false prediction. Considering T_i and F_i, we construct the pre-pruning mechanism in the optimal node-splitting control, which is depicted as a flow diagram in Fig. 5.

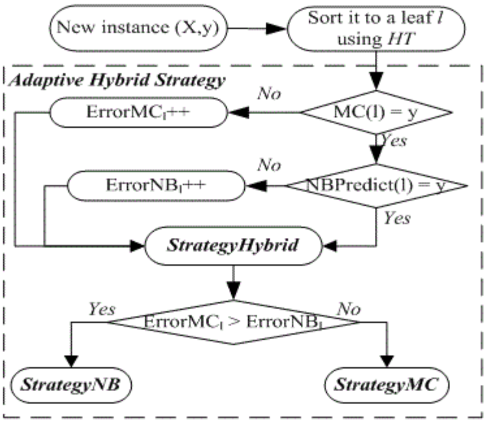


Fig. 4. Flow diagram of the Hybrid Strategy that shifts between the Majority Class and the Naïve Bayes

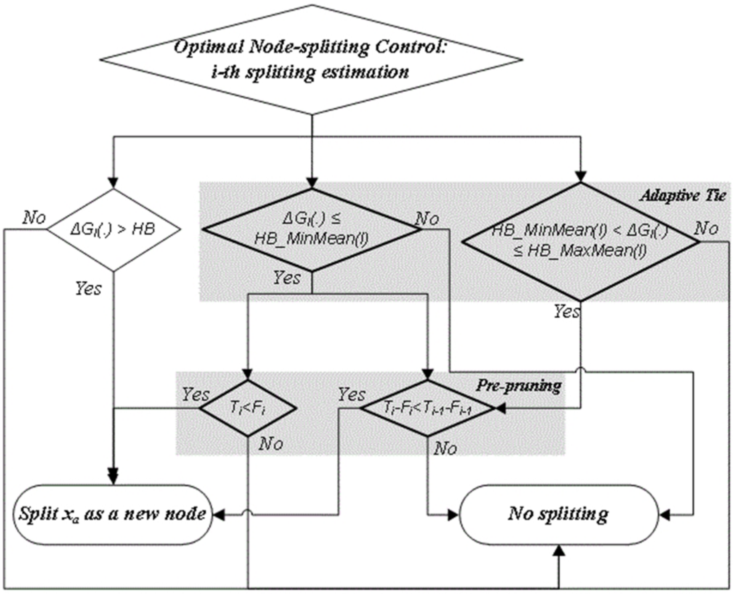

Optimal Node-splitting Control: i-th splitting estimation
Adaptive Tie
No
$\Delta G_l(.) > HB$
Yes
$\Delta G_l(.) \leq$ HB_MinMean(l)
No
Yes
HB_MinMean(l) < $\Delta G_l(.)$ ≤ HB_MaxMean(l)
No
Yes
Pre-pruning
Yes
$T_i<F_i$
No
Yes
$T_i-F_i<T_{i-1}-F_{i-1}$
No
Split x_a as a new node
No splitting

Fig. 5. Flow diagram of the optimal node-splitting control

4. EXPERIMENTS

The experiments are designed to comparatively evaluate the performance of traditional data classification algorithms and stream-based classification algorithms. In the first part, C4.5, which is a classical traditional decision tree algorithm, is compared with VFDT, which is a pioneer version of stream-based classification algorithm. Specifically, we investigate the suitability of each type of classification under biomedical data. The experiments are then moved on to the next stage where VFDT, OVFDT, and other popular variants of VFDT are put under test vis-à-vis. As a new generation of stream-based classification, VFDT has evolved into different versions, each of which has some unique features and where OVFDT is one of them. It is helpful to know which one is more suitable than the others in different situations. For all the experiments, two groups of datasets are used, normal structured datasets, which are to be used as comparison reference, and biosignal data streams. We use life science datasets that have been obtained from the real world to simulate the two types of knowledge discovery processes based on classification - one by traditional classification, the other type by stream-based classification. Those datasets can be downloaded from the UCI Data Repository (http://archive.ics.uci.edu/ml). which is a popular data repository for benchmarking the efficacy of data mining algorithms.

A customized JAVA based simulation program was written for conducting the experiments. The simulation system adopts the WEKA J48 C4.5 decision tree classifier to simulate traditional classification, and a family of MOA (Massive Online Analysis) Hoeffding Tree algorithms to simulate stream-based classification. The data stream in experiment is stored in an ARFF file format. The development environment is under JAVA JDK 1.5 and WEKA 3.6 and the system runs on a workstation of Windows 7, with 64-bits with an Intel Quad 2.83 GHz CPU and 8Gb RAM.

Table 1. UCI Experimental Raw Dataset used for Evaluating C4.5 and VFDT

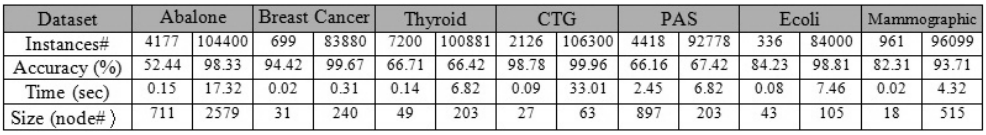
Name	Attribute#	Instances#	Type	AddIns#
Abalone	8	4177	Numeric	104400
Breast-Cancer	10	699	Nominal	83880
Thyroid	21	7200	nominal	100881
CTG	23	2126	Mix	106300
PAS	169	4418	Numeric	92778
Ecoli	8	336	Numeric	84000
Mammographic	6	961	Mix	96099

4.1 Traditional and Stream-based Classification - C4.5 vs VFDT

The first group of experiments is to simulate the traditional classification algorithm by using the WEKA J48 C4.5 classifier to construct the decision tree model. In addition to simulating the condition for large data volume, the raw dataset is also enlarged to nearly 100,000 instances by a random variable generator, so as to simulate a large volume of data stream for fair comparison.

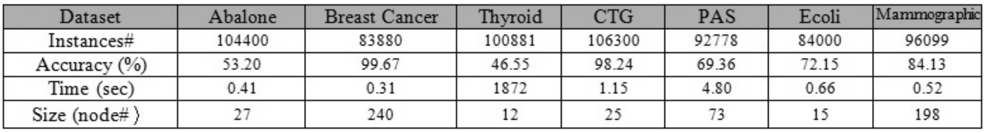
In analyzing these datasets, we observed that the C4.5 algorithm has the best accuracy in breast cancer and CTG datasets; both of which have a moderate number of attributes. It is also noticeable that the computation time required is directly proportional to the number of instances. The more instances the number is, the longer the time is spent on data mining. Amongst these datasets, PAC data has the greatest number of attributes. As a result, mining over the PAC data-

Table 2. Experimental Results by Traditional Classification



Dataset	Abalone		Breast Cancer		Thyroid		CTG		PAS		Ecoli		Mammographic	
Instances#	4177	104400	699	83880	7200	100881	2126	106300	4418	92778	336	84000	961	96099
Accuracy (%)	52.44	98.33	94.42	99.67	66.71	66.42	98.78	99.96	66.16	67.42	84.23	98.81	82.31	93.71
Time (sec)	0.15	17.32	0.02	0.31	0.14	6.82	0.09	33.01	2.45	6.82	0.08	7.46	0.02	4.32
Size (node#)	711	2579	31	240	49	203	27	63	897	203	43	105	18	515

Table 3. Experimental Results by Stream-based Classification



Dataset	Abalone	Breast Cancer	Thyroid	CTG	PAS	Ecoli	Mammographic
Instances#	104400	83880	100881	106300	92778	84000	96099
Accuracy (%)	53.20	99.67	46.55	98.24	69.36	72.15	84.13
Time (sec)	0.41	0.31	1872	1.15	4.80	0.66	0.52
Size (node#)	27	240	12	25	73	15	198

set by a C4.5 algorithm produces a relatively poor accuracy. The experiments results are tabulated below. Accuracy is defined as the number of correctly classified instances over the total number of instances. The accuracy was the final performance measured at the end of each experimental run.

In contrast to traditional classification, the accuracy of the decision tree in stream-based classification is accumulative, which is increasing with more and more instances being processed. However, the same experiment run by C4.5 shows that when the decision tree in traditional classification matures (after being trained with a considerable number of instances) further data processing just consumes more time. Comparing Table 2 with Table 3, we can see that one of the disadvantages of traditional classification is a long computation time required. In other words, when more instances arrive, the time spent on rebuilding the tree model becomes longer and longer. Considering a real-time scenario where the model is required to update in each single second, traditional classification fails to construct a decision tree model within a short amount of time.

In our previous research [8], a general comparison between the C4.5 algorithm and VFDT was carried out. The experiments are highlighted as follows: (1) used medium synthetic dataset to simulate a traditional C4.5 decision tree algorithm and VFDT. The result shows that C4.5 can achieve a higher accuracy than VFDT in a medium dataset, but that VFDT operates in a faster computation time and smaller tree size than C4.5. (2) Used a real world small dataset to compare C4.5 and VFDT stream mining. The result is similar to the medium synthetic dataset. And (3) C4.5 reveals its limits in handling a huge dataset. Both nominal and numeric synthetic datasets of huge sizes are used with VFDT. Simulation results find that VFDT accuracy is sensitive to noise data. Tree size linearly increases when more instances arrive. Numeric dataset results in a more complex tree model, but it is more accurate than nominal data in VFDT. From the experiments results done here, we find that: C4.5 has a higher accuracy but is slower in running time because of the multi-scans over a database. Stream-based classification using VFDT has a very fast speed but achieves a relatively lower accuracy in the case of small data size. The time for the booting step of VFDT is long since the algorithm grows from a very low accuracy at the start.

4.2 Evaluation of Model Usefulness

The following experiment was done on evaluating the usefulness of traditional and stream-based classification. In a real-life environment, for example in a healthcare center, where a predictive is used as a major function in its decision support system, the sequence of operation usually goes by first building up a decision-making model with an underlying structure of a decision tree, and then it is put into use along with the incoming data, which is similar to testing for accuracy in our experiment. Decisions were made in real-time by the system built by traditional classification, and the model is supposed to be good until awhile later when the model needs to be updated by rebuilding the decision tree with the inclusion of the new data. This process will essentially affect the usefulness of the classification model. Usefulness is defined as a mix of how long a decision model will remain useful before a significant decline in accuracy, how often the updates should be required, and how much/little the interruptions would impact the availability of the system. Another experiment is set out to verify the usefulness of the data mining algorithms under this type of working sequence. This is different from the previous experiments in which the instances are entirely input into the data mining programs all at once (at each testing point along the x-axis); the divisions of the data for model training, cross-validation. and testing were automatically done by the programs according to the default settings.

For the breast cancer dataset, C4.5 is used to construct the decision tree model by updating the underlying rules at recurring intervals when every 400 new instances have arrived. As a result, the experiment presents the first three periods during which the model update took place. In each period, the first 400 pieces of data are collected for rule building, while the other 1,200 pieces of data are used for prediction by the just updated decision-making model. The simulated result for the case of C4.5 is shown in Fig. 6. Clearly, the established rules by the aged data fall short of accuracy for making predictions with the new coming data. This is reflected by similar declining trends over the three periods of time. Comparatively, we applied the same dataset for VFDT in another experiment.

The result in Fig. 7 shows that the performance curve is rather steady (in contrast with the downward lines that were broken up as in C4.5) and the general accuracy is ever improving as the decision tree model gets updated by the unique mining mechanism of VFDT, each time when new data is fed in. However, VFDT is an accumulative algorithm where the accuracy is increasing when more and more instances come. In other words, the booting step of VFDT may be longer than that of C4.5 to some extent. It is also a disadvantage of VFDT compared to traditional classification algorithms. A remedy called the VFDT-boot has been proposed in [9] for tackling the long booting time for the stream-based model to reach a satisfactory level of accu-

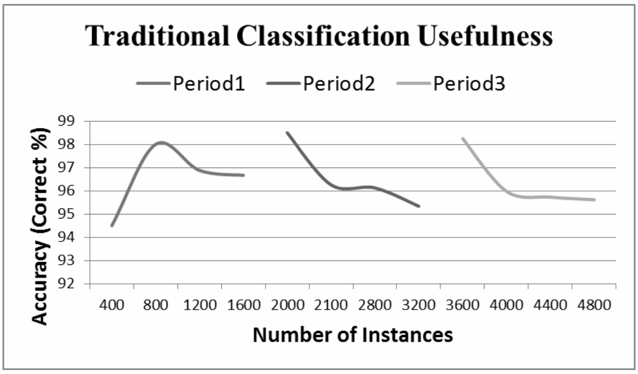


Fig. 6. Usefulness of the Traditional Classification Model

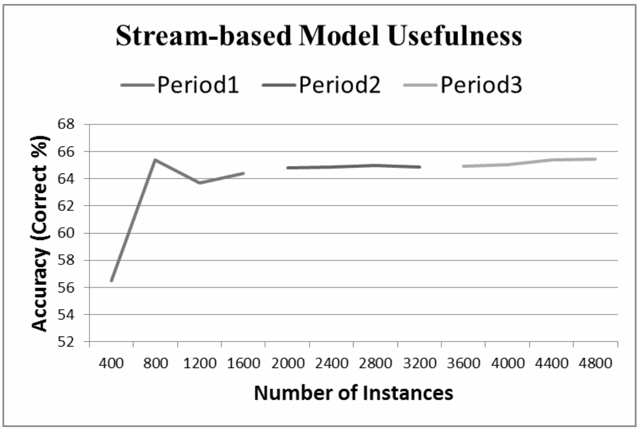


Fig. 7. The Usefulness of the Stream-based Classification Model

racy at the beginning. The VFDT-boot is VFDT bootstrapped with an over-pruned version of the tree produced by C4.5. The accuracy of the VFDT-boot has an increase gain between 48% and 89%. The first part of 19,000 data instances was used to build a pruned C4.5 decision tree model. The VFDT is then run on the top of that pre-trained model since their decision trees are compatible in data structure. VFDT-C45Boot obtains a better accuracy than VFDT alone.

4.3 Stream-based Classification - OVFDT vs. VFDT and Others

The purpose of the remaining experiments here is to evaluate the performance of a number of stream-based classifiers such as VFDT, Additional Option Trees, Bagging in comparison of OVFDT integrated with a Functional Tree Leaf (which was innovated by the authors) under the environment of a biosignal data streams analysis. It was already demonstrated in earlier work [10] that OVFDT outperformed VFDT. However, it is yet to be verified whether further gains in performance could be observed when the Functional Tree Leaf made modifications in the prediction phase. Furthermore, it is important to validate the feasibility of using OVFDT+Functional_Leaf with respect to biosignal stream classification. Our performance evaluation covers three aspects from the perspective of a real-time biosignal classifier: (1) comparison of accuracy, which is defined as the percentage of correctly classified instances; (2) comparison of the model tree size, which is the amount of leaves and nodes that would be stored in the runtime memory of the computer; and (3) comparison of the computation time measured by processing per ten thousands instances of data stream, which is interpreted as the average of the learning time plus testing time. The same datasets are used to train and test all the decision tree models. The experimental data are real-life datasets that were acquired from the UCI repository. Two datasets of structured record types of classification and two datasets of stream-based types of biosignals, namely EEG (Electroencephalogram) and ECG (Electrocardiogram), are deliberately chosen. The feasibility of the stream-based algorithms would be put under test in mining through these two types, with a total of four datasets. In particular the biosignal dataset are described in Table 4.

The original VFDT serves as a benchmarking reference here for testing the performance of OVFDT. The experiment parameters are chosen as follows: the tie-breaking threshold is set at

Table 4. Biosignal datasets used in the stream-based experiment

EEG	12 technical variables describe the heart rates and situational information about the heart conditions (e.g. measure of contractility around the heart, movements of the segments of the left ventricle, etc.). Total: 132 instances
Target	To predict whether or not the patient will survive for at least one year.
ECG	Fetal cardiotocograms were generated with their respective diagnostic attributes measurements. 23 meaningful features were selected by 3 experienced obstetricians. They assigned class labels by their knowledge for each vector, at a total of 2,126.
Target	Features describe one of the 10 different morphologic patterns that are used to predict 3-classes fetal states.

0.5; 200 instances to be observed before a split attempts at a leaf; the splitting confidence is at 106; and the information gain is used as a measure for splitting the evaluation. As observed from the results as shown in the following radar charts, OVFDT and the other variants generally yield better than those of VFDT in terms of accuracy. The overall accuracy of OVFDT+Functional_Leaf is almost as good as those of HOT and Bagging. However, they do not get compromised by tree sizes. HOT is a type of VFDT that produces additional option nodes so that several tests can be applied concurrently. As a side effect, HOT leads to separate paths by growing additional multiple trees at each node. A testing sample can traverse down different optional paths of the tree that leads to yielding more alternatives to find an answer. As expected, HOT and its variants required the largest tree sizes because of a heavy cost to improve accuracy. This problem is aggravated for biosignal data, as the tree size for HOT is almost 10 to 17 folds larger than OVFDT in the case of EEG. HOT_ADA has an improvement over HOT: each leaf stores an estimation of the current error, so it adaptively avoids the erroneous paths. But according to the results, HOT_ADA did not show any significant advantage over the other kinds of HOT for biosignal data. It takes the longest running time in general except for ECG. Lastly the state-of-the-art in data mining is Bagging, which is an ensemble method of classification, and voting is used to decide which path/sub-tree to follow. The method we experimented with is known as the Adaptive-Size Hoeffding Tree (ASHT). It is a kind of HOT, but it has an additional advantage by trimming off some nodes to reduce tree size when the number of split

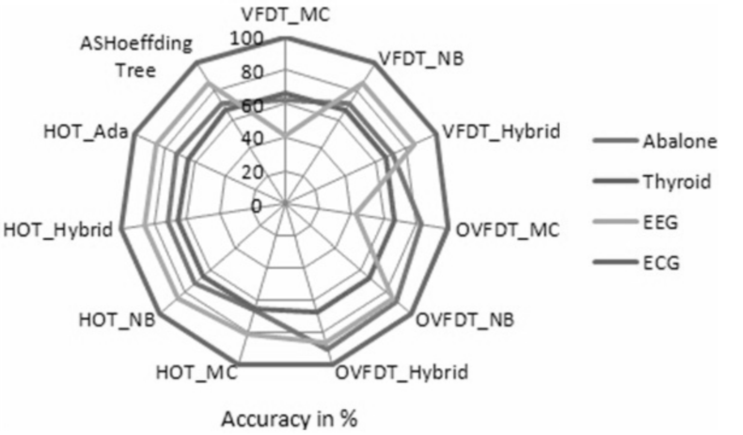


Fig. 8. Accuracy comparison of various VFDT, OVFDT and variant algorithms

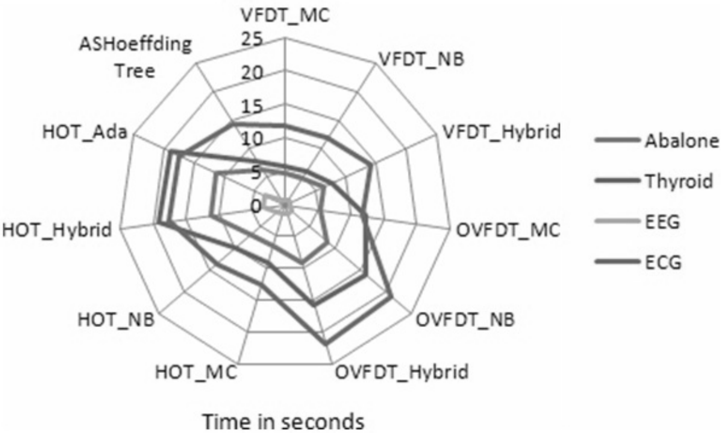


Fig. 9. Time comparison of various VFDT, OVFDT and variant algorithms

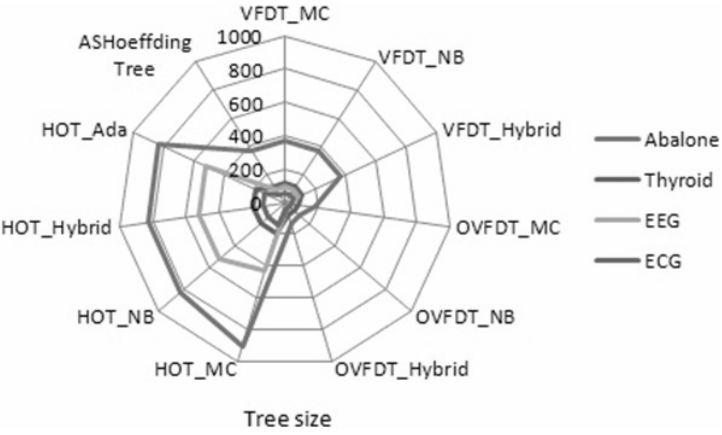


Fig. 10. Tree size comparison of various VFDT, OVFDT and variant algorithms

nodes exceeds a threshold. It however was outperformed by OVFDT in accuracy, and the tree size that ASHT achieved is only as good as VFDT by using the ensemble method. In summary OVFDT and its variant by Functional Leaf succeed in terms of better accuracy than other types of stream-mining classifiers and the reduction in tree size is always significantly larger. However, the apparent drawback is the extra processing time in controlling the node-split task by computing the adaptive tie-threshold and pre-pruning for OVFDT. The probabilities calculations by Naïve Bayesian at the Functional Leaf consumes quite some time too for the enhancement of prediction accuracy for OVFDT. The overhead in running time is about 10% to 30% when compared to VFDT.

5. Conclusion

Biosignal classification has a long history in the research community. Recently some researchers are looking at clinical decision support systems as real-time monitoring and automated applications. Though this research direction is rising, the underlying classification algorithm has not been clearly defined, especially for real-time and mobile biosignal systems. By reviewing the current literature most, if not all, of the past papers centered on traditional data mining methods such as neural networks, SVM, C4.5 etc., which are doubted to be able to cope with mining continuous data streams. On the other hand, a new generation of data stream mining algorithms has been recently invented, with the specific purpose of efficiently handling moving data streams. Therefore, in this paper, we conducted a pioneer investigation on the choices of biomedical classification algorithms for analyzing biosignal data streams. Both traditional and stream-based classification algorithms are put to test in simulation for comparatively evaluating their performances. C4.5 was chosen as a classical algorithm representing a traditional classification algorithm; VFDT, the benchmark in stream-based classification algorithms, an optimized version called OVFDT plus OVFDT integrated with Functional Leaf, and a collection of lately innovated VFDT variants such as HOT, Bagging, etc. were tested together in the performance comparison.

So far the type of decision tree classification learning belongs to supervised learning where samples must be labeled in the training process. This consequently means a preprocessing step is necessary that assigns result labels to each window of the data stream segment. However, the VFDT takes testing and training at the same time - when arriving data stream segments carry class labels and training takes place. Otherwise, when a data stream segment is not tagged with any label, VFDT automatically assumes its testing mode and predicts a class for that unlabeled data stream segment by its trained tree-node structure.

By comparing C4.5 and VFDT, which respectively represents traditional and stream-based classification, C4.5 suffers a number of shortcomings: C4.5 needs to periodically update its trained model in batch mode so that its tree structure reflects the underlying patterns from the latest arriving data. The periodic updates mean interruptions to the availability of the decision tree. The time taken for each update grows in proportion to the size of the database and the whole database that includes the new data would need to be scanned repeatedly for the update. Fundamentally this is the bottleneck of the traditional decision tree methods because it requires rebuilding the tree model by accessing the whole set of data all over again for each update. In contrast, a stream-based classification model like VFDT and its variant only requires one-pass learning from the data, and they are able to update the decision tree incrementally as new data comes in without referring back to the seen data.

Out of the family of VFDT, OVFDT continues to show its superiority for biosignal classification. By its characteristics of being able to achieve optimal balance between good accuracy and compact tree size, OVFDT handles well the potentially infinite biosignal data streams without incurring tree size explosion. All of the concerns of the memory constraints, speed, and classification accuracy are met, making OVFDT a suitable classification model for stream mining biosignals.

REFERENCES

[1] S. Fong, H. Yang, “The Six Technical Gaps Between Intelligent Applications and Real-Time Data Mining: A Critical Review,” *Journal of Emerging Technologies in Web Intelligence* (JETWI), Academy Publisher, ISSN 1798-0461, Vol.30, No.2, 2011, pp.63-73.

[2] Q. Fang, F. Sufi, I. Cosic, *A Mobile Device Based ECG Analysis System, Data Mining in Medical and Biological Research*, In-Tech, Vienna, Austria, 2008, pp.320-338.

[3] H. Hermens, V. Jones, "*Extending Remote Patient Monitoring with Mobile Real Time Clinical Decision Support*," *Proc. of IEEE-EMBS Benelux Chapter Symposium,* 2009, Enschede, The Netherlands, pp.50-53.

[4] M. Zwaag van der, E.L. Broek van den, J.H. Janssen, "*Guidelines for biosignal driven HCI," Proc. of ACM CHI 2010 Workshop - Brain, Body, and Bytes: Physiological user interaction,* 2010, Atlanta, GA, USA, pp.77-80.

[5] H.G. Lee, K.Y. Noh, K.H Ryu, "*Mining Biosignal Data: Coronary Artery Disease Diagnosis Using Linear and Nonlinear Features of HRV," PAKDD 2007 Workshops,* LNAI 4819, 2007, pp.218-228.

[6] J.C.Y. Chen, V.S.M. Tseng, An Integrated Bio-Signal Data Mining Mechanism with Applications on Asthma Monitoring and Prevention, [dissertation], MSc Thesis, National Cheng Kung University, Taiwan, 2007.

[7] P. Domingos, G. Hulten, "*Mining high-speed data streams*," *Proc. of KDD 2000*, New York, USA, 2000, pp.71-80.

[8] H. Yang, S. Fong, "*An Experimental Comparison of Decision Trees in Traditional Data Mining and Data Stream Mining*," *The 6th International Conference on Advanced Information Management and Service (IMS 2010)*, 30 November - 2 December, 2010, Seoul, Korea, pp.442-447.

[9] H. Yang, S. Fong, A. Ip, S. Mohammed, "*Case-based and Stream-based Classification in Biomedical Application*," *The Eighth IASTED International Conference on Biomedical Engineering (Biomed 2011)*, 16-18 February 2011, Innsbruck, Austria, pp.207-214.

[10] H. Yang, S. Fong, "*Optimized Very Fast Decision Tree with Balanced Classification Accuracy and Compact Tree Size*", *Proc. of the 3rd International Conference on Data Mining and Intelligent Information Technology Applications (ICMIA2011)*, IEEE Press, 24-26 October, 2011.

Simon Fong

He graduated from La Trobe University in Australia, with a First Class Honours BEng. Computer Systems degree and a PhD. Computer Science degree in 1993 and 1998 respectively. Simon is now working as an Assistant Professor in the Computer and Information Science Department of the University of Macau. He is also one of the founding members of the Data Analytics and Collaborative Computing Research Group in the Faculty of Science and Technology. Prior to joining the University of Macau, he worked as an Assistant Professor in the School of Computer Engineering at Nanyang Technological University in Singapore. Before his academic career, Simon took up various managerial and engineering posts, such as being a systems engineer, IT consultant, integrated network specialist, and e-commerce director in Melbourne, Hong Kong. and Singapore. Some companies that he worked at before include Hong Kong Telecom, Singapore Network Services, AES Pro-Data, and the United Overseas Bank in Singapore. Dr. Fong has published over 140 peer-reviewed international conference and journal papers, mostly in the area of E-Commerce technology, Business Intelligence, and Datamining.

Yang Hang

He is a PhD candidate at the University of Macau. He obtained an MSc (First Honor) in Electronic Commerce Technology from the University of Macau in 2009; and a Bachelor's degree in Economics and Electronic Commerce from Guangdong University of Foreign Studies (China) in 2007. He worked for the companies of Fortis Insurance and China Petrol in Hong Kong and Beijing. Dr. Simon Fong supervises him and his research interests cover data mining, business intelligence, electronic commerce, and web intelligence. So far, he has over 20 publications, including journals and conference papers.

Sabah Mohammed

Dr. Mohammed received his B.Sc. in Applied Mathematics (Baghdad University 1977), and his graduate degrees in Computer Science from Glasgow University (MSc 1981) and Brunel University (PhD 1986). Since late 2001, Dr. Mohammed has been an Associate Professor of Computer Science at Lakehead University. Formerly, from 1986-1995, Dr. Mohammed was an Assistant/Associate Professor of Computer Science at Baghdad University holding the position of being the Graduate Organizer in Computer Science. During 1996-2001, he served as the Chair of Computer Science at the following four different universities: Amman University (1995-1996), Philadelphia University (1996-1997), the Applied Science University (1997-2000), and the Higher College of Technology (2000-2001). Dr. Mohammed has co-authored four text books in Compilers, Artificial Intelligence, Java Programming and Applied Image Processing. He has published over 70 refereed publications, was the MSc advisor for 17 students and 1 PhD student, and has received research support from a variety of governmental and industrial organizations. Dr. Mohammed is a member of the British Computer Society, a member of the Canadian Image Processing & Pattern Recognition Society, and a Member of the IEEE Signal Processing Society.

Jinan Fiaidhi

Dr. Jinan Fiaidhi has been a full Professor and the Graduate Coordinator with the Department of Computer Science at Lakehead University in Ontario, Canada since late 2001. She is also an Adjunct Research Professor with the University of Western Ontario. She received her graduate degrees in Computer Science from Essex University (PhD 1983) and Brunel University (PhD, 1986). During the period of 1986-2001, Dr. Fiaidhi served at many academic positions (e.g. the University of Technology [Assoc. Prof and Chairperson], Philadelphia University [Assoc. Prof], Applied Science University [Professor], Sultan Qaboos University [Assoc. Prof.]). Dr. Fiaidhi's research is focused on mobile and ubiquitous learning that utilizes emerging technologies. Dr. Fiaidhi's research is supported by the major research granting associations in Canada (e.g. NSERC, CFI). Moreover, Dr. Fiaidhi is a Professional Software Engineer of Ontario (PEng), a Senior Member of IEEE, a member of the British Computer Society (MBCS), and a member of the Canadian Information Society (CIPS) holding the designate of ISP. Dr. Fiaidhi has intensive editorial experience (e.g. Editor of “IEEE IT-Pro,” and being the Associate EiC of the “Journal of Emerging Technologies in Web Intelligence”).