White Learning Methodology: A Case Study of Cancer-Related Disease Factors Analysis in Real-time PACS Environment

Tengyue Li, Simon Fong[*], Shirley W. I. Siu
Department of Computer and Information Science
University of Macau, Macau SAR
{mb75436, ccfong, shirleysiu}@um.edu.mo

Lian-Sheng Liu[*]
Department of Radiology
First Affiliated Hospital of Guangzhou University of TCM, China
llsjnu@sina.com

Xin-she Yang
Department of Design Engineering and Mathematics
Middlesex University, London, UK
X.Yang@mdx.ac.uk

Sabah Mohammed
Department of Computer Science
Lakehead University, Thunder Bay, Canada
mohammed@lakeheadu.ca

Abstract

Background and Objective: Bayesian network is a probabilistic model of which the prediction accuracy may not be one of the highest in the machine learning family. Deep learning (DL) on the other hand possess of higher predictive power than many other models. How reliable the result is, how it is deduced, how interpretable the prediction by DL mean to users, remain obscure. DL functions like a black box. As a result, many medical practitioners are reductant to use deep learning as the only tool for critical machine learning application, such as aiding tool for cancer diagnosis.

Methods: In this paper, a framework of white learning is being proposed which takes advantages of both black box learning and white box learning. Usually, black box learning will give a high standard of accuracy and white box learning will provide an explainable direct acyclic graph. According to our design, there are 3 stages of White Learning, loosely coupled WL, semi coupled WL and tightly coupled WL based on degree of fusion of the white box learning and black box learning. In our design, a case of loosely coupled WL is tested on breast cancer dataset. This approach uses deep learning and an incremental version of Naïve Bayes network. White learning is largely defied as a systemic fusion of machine learning models which result in an explainable Bayes network which could find out the hidden relations between features and class and deep learning which would give a higher accuracy of prediction than other algorithms. We designed a series of experiments for this loosely coupled WL model.

Results: The simulation results show that using WL compared to standard black-box deep learning, the levels of accuracy and kappa statistics could be enhanced up to 50%. The performance of WL seems more stable too in extreme conditions such as noise and high dimensional data. The relations by Bayesian network of WL are more concise and stronger in affinity too.

Conclusion: The experiments results deliver positive signals that WL is possible to output both high classification accuracy and explainable relations graph between features and class.

Keywords—Data mining methodology, deep learning, Bayesian network, Radiological data analysis.

I. INTRODUCTION

Nowadays, many technology companies have begun to study the use of AI for medical diagnosis, especially for the diagnosis of cancer. The detection ability helps reduce the time taken for clinicians to examine pathology scans. This achievement encouraged the development of large-scale online computing platform which allows huge and high-resolution pathology images to be stored, retrieved and analyzed as per demanded by the medical professionals. The information system is recently evolving into big data platform that is usually distributed with cloud storage and access, providing on-demand image uploading and downloading services between the central data repository and the end-user node, right to the doctor’s desktop. With this large-scale diagnostic information system in place, pathologists are well connected to the data, the analytic services and other medical professionals for diagnostic tasks, group discussion and decision making, associated with radiation and chemotherapy could be supported very well by ICT. Given the sheer volumes of data, requirements for latency constraints for real-time and accurate and timely detection by machine learning, the information system is indeed an epitome of extreme automation. This big data and large-scale cancer imaging and diagnostic platform would be deployed to serve clinicians, medical professionals, radiologists, and so on, at various levels, with a wide geological coverage. For example, national cancer big data platform would serve a hierarchy of national research labs, both government and private hospitals, specialized cancer treatment centers, across different provinces, cities and suburbs. Such large-scale information system poses great computational challenges in both strong hardware infrastructure and software performance requirement.

One of the most important parts of AI medical assistant is the powerful machine learning algorithm which could accurately select the case with cancer from a group of cases. Google has announced that the deep learning tool outperformed traditional prediction models in terms of speed and accuracy. Although accurate classification from deep learning helps a pathologist's microscopic examination of a tumor in patients for cancer diagnosis, making treatment decisions are far more than just detecting the presence of nodal metastasis. In the Google blog[1], it was mentioned:

"*While LYNA achieved significantly higher cancer detection rates (Liu et al. 2017) than had been previously reported, an accurate algorithm alone is insufficient to improve pathologists' workflow or improve outcomes for breast cancer patients. For patient safety, these algorithms must be tested in a variety of settings to understand their strengths and weaknesses. Furthermore, the actual benefits to pathologists using these algorithms had not been previously explored and must be assessed to determine whether or not an algorithm actually improves efficiency or diagnostic accuracy*."

That implies a deep learning algorithm that is fast and accurate in a real-time PACS environment may not be the only criteria. When the results of machine learning can affect a patient's life or death, the use of deep learning should be more thoughtful. Doctors prefer to see more supporting evidences and more comprehensive information regarding how a prediction is derived, instead of just being told by a machine like a black box about a computed outcome. On top of a predictive model which outputs generated prediction, doctors opt for a number of visualization tools and reports for them to inspect and analyze. This is the motivation for proposing a hybrid black and white box machine learning model, which has the benefits of both – highly accurate prediction and interpretable models for explaining how the predicted results came by, yet suitable for operating in an incremental machine learning environment that meets the real-time demands of PACS-based decision support.

It can be safely concluded that a highly accurate predictive model is only a part of the AI medicine strategy. Many recent machine learning research endeavors and commercialization developments are geared towards using advanced machine learning technologies, such as deep learning. However, there are more than just about predicting a medical verdict by the machines; software tools that empower doctors to analyze and interpret the data & results could be equally if not more important for clinical decision supports. A typical medical analysis infrastructure is shown in Figure 1.

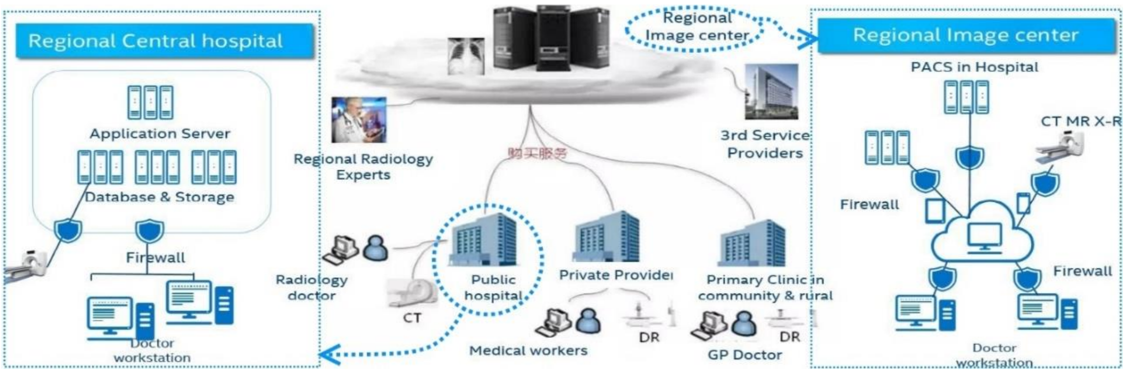


Fig. 1. The infrastructure of medical AI diagnostic big data system.

The infrastructure is an epitome of large-scale medical AI diagnostic big data system. All the imaging and medical record data are meant to be stored and managed by a central cloud which typically should be owned and supported by the national government. Heterogenous data are continuously uploaded from regional image centre that has a variety of PACS, with imaging data from CT scans, X-ray, and MRI etc., that are generated from patients on a daily basis. All such data are centralized at a Cloud, which in turn offers query services to public/private hospital/clinics, for medical diagnosis and inquiries.

This type of large-scale medical information evolves from Cloud-based PACS, which was designed to replace the need to store and manage hard-copy films and reports in space-consuming shelving and rooms. Instead, medical images and non-image data can be securely stored digitally on premises or in the cloud. Cloud-based PACS store and back up an organization's medical imaging data to a secure off-site server. A cloud PACS enables medical staff to view medical imaging data from any approved devices, such as a smartphone, as an online service.

most of the services hinge on the quality of the data analytics and reasonable fast service turnaround time. These requirements are typically the three criteria above-mentioned: accuracy & reliability from the data analytics, real-time accessibility and interpretable data or disease analysis, they are needed for supporting useful remote medical services in addition to the basic system reliability requirement at the data networking and technical system levels.

[1] https://ai.googleblog.com/2018/10/applying-deep-learning-to-metastatic.html

[*] Co-correspondence authors

The need for data analytics in such large-scale and distributed medical data environment is the motivation for this thesis study. We focus on devising a machine learning methodology that embraces incremental learning, accurate and fast model training, and results of data analytics that can be interpreted easily by medical practitioners. The first two criteria can be met by providing an incremental learning solution. This is where the AI part usually comes in, by provisioning some intelligent advices and/or decision supports in the form of machine generated predictions and reasoning results. As mentioned earlier, there are reported news about recent sentiments from doctors towards AI enabled medical diagnosis and prediction explicitly stressing that, doctors prefer to see more supporting evidences and more comprehensive information regarding how a prediction is derived, instead of just being told by a machine like a black box about a computed outcome. Figure 2 shows a combination of hardware and software that form the building blocks of the large-scale medical system. At the software side, on top of a predictive model which outputs generated prediction, doctors opt for a number of visualization tools and reports for them to inspect and analyse. This is the motivation for proposal a hybrid black and white box machine learning model, which has the benefits of both – highly accurate prediction and interpretable models for explaining how the predicted results came by.

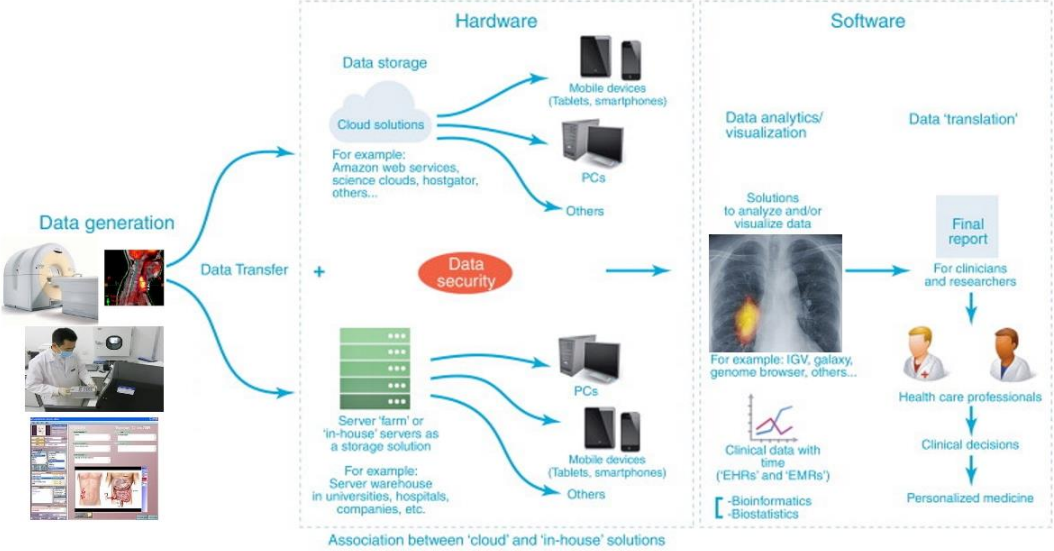


Fig. 2. A combination of hardware and software building blocks.

II. WHITE LEARNING MODEL

A. Related Work

Nowadays, cancer detection by Deep Learning is a hot topic especially in early detection. It can improve the survival rate in long term. Most of the detection work by Deep Learning based on medical images which is useful in early detection and monitor after cancer treatments. Traditionally, checking medical images is by human. Manually checking cannot avoid careless mistakes especially facing to these tons of numbers of medical images. So, Deep Learning applied in cancer detection changes the situation. Therefore, from early 1980s, computer-aided diagnosis (CAD) systems were introduced to assist doctors in interpreting medical images to improve their efficiency [1]. In CAD system, it basically used machine learning which the feature extraction is important to. So, in a long time, feature extraction is hot research topic. Depending on different kind of cancers, there are specific ways of feature extraction. While, methods adopting feature extraction have limitations. To further improve CAD system, more and more researchers turn to study Deep Learning which is representation learning techniques that learns hierarchical feature representation from image data. Combining with GPU, it had a great achievement in cancer detection and diagnosis.

It is noticed that deep learning techniques which are strong in image recognition, have been applied in detecting, classifying and segmenting on various medical domains [2]. Most of the works however are focused on radiological imaging classification. Quite a few of them are applied on general prediction/classification. It is an emerging trend however in developing deep learning alike machine learning tools for interpretable models which could be easily understood by human users. In the past there had been some works, though very technical, about extracting weights from the convolution neural networks. From the internal information extracted from the convolution layers of the neural networks, one can infer about the importance of the features pertaining to the predicted target. This requires specialized computing skills, which may be quite difficult for general medical practitioners.

On the other hand, Bayesian Artificial Neural Network [3][4][5] has been formulated for trying to offer both good level of prediction and probabilistic inference inherent by Bayes Network. It is based on a profound foundation that Bayes interference is capable of detailing a whole probability distribution over possible outcomes of hypothesis h, instead of a single predicted value of h. The golden Bayes' rule computes the posterior probability of h given the facts, f, is $p(h|f) = \frac{p(f|h)p(h)}{p(f)}$ where $p(h)$ is the prior probability h before knowing the facts f; and $p(f|h)$ is the likelihood of having such facts f given h. As a tightly coupled computational method, the entire probability distribution of the Bayes Network is applied to the neural network over possible outcomes of hypothesis. The probabilities form up the neural network weights *w* given the training dataset, *p*(*w*|*f*). Based on the given weights of values from the Bayesian inference, the results are a posterior distribution over a possible set of different configurations (sized) neural networks and their outputs. So instead of limiting getting the prediction from a single value, we could obtain possible answers from an entire distribution from a collection of different sized networks which were built from Bayesian posterior distribution. This design inspired further researchers to extend the neural network to convolutional neural network. For example, a Bayesian Deep CNN was proposed [6] for learning features by Gaussian networks. It is designed for capturing higher-order features in a case of text mining. From the training dataset, network motifs are derived and used for pre-training the weights of CNN for enhancing its discriminative power. These models work well, despite of their relatively high complexity

Some researchers however turn to some simpler fusion models, regarded as semi-coupled white learning model in the context of our white learning framework It is characterized by building two or more machine learning models in the form of Bayesian or similar white box decision tree model, and deep learning or neural network and the like, by transfer learning. Information or prior knowledge learnt from one model is passed onto the other model, in such a way that the learned outcomes could benefit the construction of another model. In semi-coupled white learning model, either control, knowledge, or messages which are related to shaping up the machine learning model are passed from one model to another. Several attempts were made in the following combinations:

- Hidden Markov Model + Recurrent Neural Network or Long-short-term-memory Neural Network [7]
- Backpropagation Neural Network + Bayesian Network [8]
- Backpropagation Neural Network + Bayesian Network + Prior knowledge infusion [9], etc.

To further simplifies the hybridization, researchers resorted to loosely-coupled model by which the connection is limited to passing data. In this model, a Bayesian Network that represents white-box learning and a Neural Network which learns like a black box are setup and operate independently. They learn collectively hand-in-hand or in sequence, passing the learnt/processed training data along the training process. Some typical approaches of data passing are as follow:

- Ensemble learning using multiple Neural Networks [10]
- Ensemble learning using Decision Tree, Support-vector-machine and two types of Neural Networks [11]
- Ensemble learning using multiple Neural Networks and multiple Bayesian Networks [12]
- Ensemble learning using a single Convolutional Neural Network and a single Bayesian Networks (naïve version without any optimization) [13]

It can be seen that there exist many possible combinations and varieties of putting different and multiple learners together; they might share and co-process certain portions of training data. Out of the mixed models, the one that offers the highest level of accuracy is selected as a winner for doing the final prediction. This approach is simple, but often the performance is not at its maximal. Therefore, in this work, a suite of White Learning model is proposed which leverages fast optimization functions suitable for PACS environment where speed, accuracy and interpretability are concerned.

B. White Learning By Misclassified Recall And Swarm Feature Selection

In this thesis study, a relatively easy to implement white learning model is proposed and setup. There are one each classical black and white learner in the system, one is NBU the other is DL, the default version with default parameter settings by the benchmarking software Weka. The two learners are connected by a filter class called Misclassified Recall which essentially cleans up the data and using NBU which is relatively fast and passing the cleaned data to DL for refined accuracy. The WL model mainly works at the data level, taking a matrix of dataset, while loading the data by a sliding window (of size 1000 instances) at a time, the optimal features are selected, and the problematic data instances are removed progressively.

In our proposed WL model, there are modification schemes for further enhancing the prediction performance made available. They are novel ideas designed for upgrading the efficacy of swarm feature selection, namely Early abandonment, Accelerated swarm and Teng-yue swarm. Those novel ideas are developed in this study; experimentation of the three novel modifications are carried out, the results are reported in subsequent sections.

The block diagram for our proposed WL model is shown in Figure 3. A glossary of terms that are pertaining to Figure 3 is defined in Appendix A for clarity. Together with the block diagrams for individual white-learner model and individual black-learner model respectively, we can compare the outputs resulted from the three models. The individual white learner model outputs both prediction result with performance measured, and a causal graph as Bayesian network. It is noted that the Bayesian network generated by individual white-learner has full number of nodes and possible relations in the network. The individual black learner model however only outputs a prediction result with performance measured. Nevertheless, the WL model, outputs not only the sum of the outputs from the individual white-learner and black-learner, the prediction accuracy of the DL in the WL model should be higher, and the Bayesian network would be more concise in terms of network structure and the quality of the causality paths.

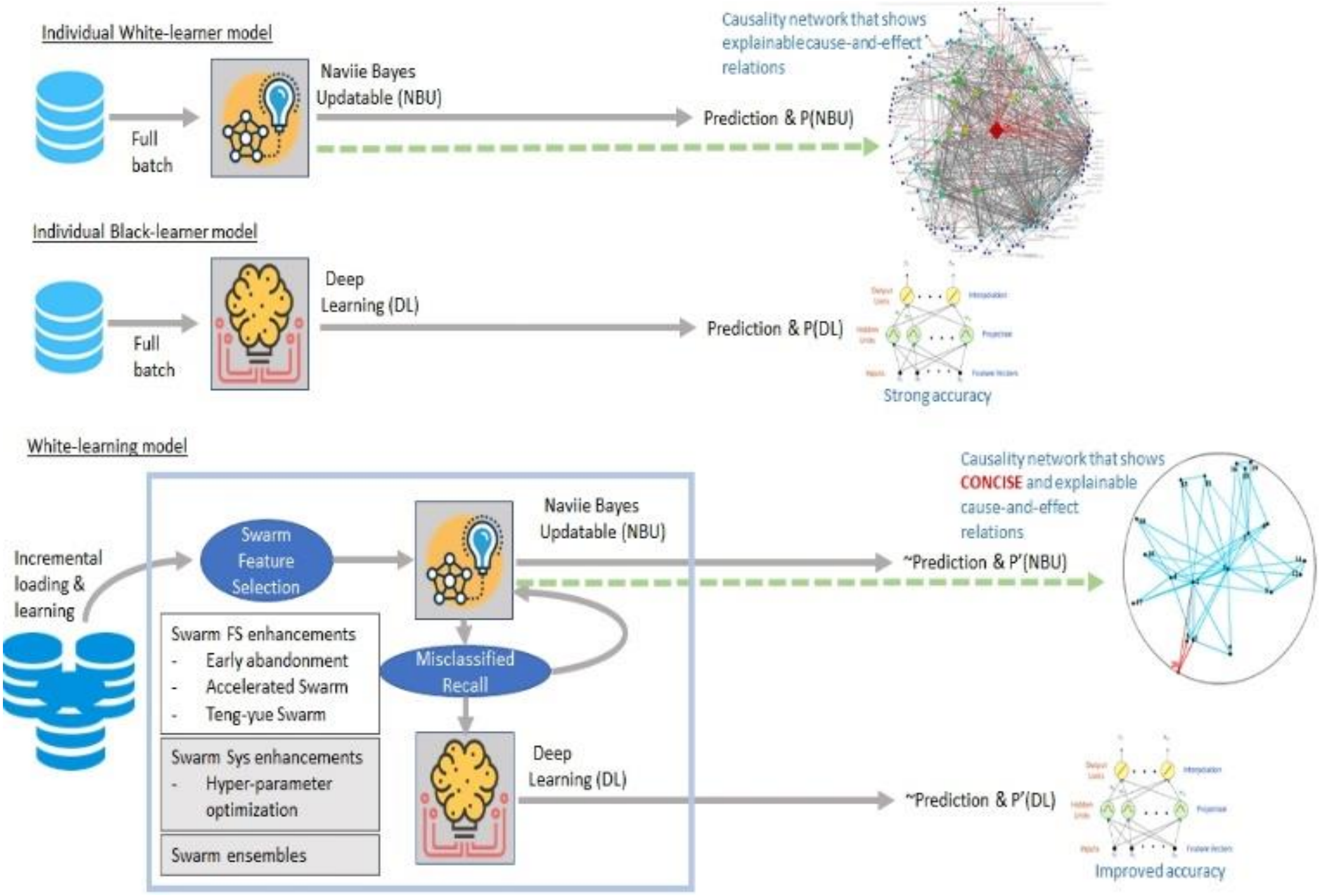


Fig. 3. (Up) Standard white learner model and black learn model. (Below) Proposed white learning model.

WL works basically by first negotiating with the users about the QoS requirements. As this WL framework is supposed to support real-time AI medicine applications, time, latency, accuracy, reliability indicators (kappa statistics) and perhaps other metrics like false alarm rate, precision and recall, etc., are of concern to medical applications. There should be a calibration phase in the WL framework where a collection of algorithms that were to be implemented as white-learner and black-learner would have been tried; their performance are recorded and retained as some knowledge cache. Therefore the WL framework would know how to find the best possible combinations of candidate algorithms with known performance that would meet the QoS requirements by the users. Once the user's QoS request is found acceptable, it proceeds to picking the best available algorithms and parameters. Start loading in the data incrementally. When an appropriately sufficient amount of data is accumulated, a dynamic data pre-processing process initiated. Based on the accumulated data that is held in a cache, the two facets of the structured dataset are fixed. Over the columns of the dataset, a stochastic feature selection is applied which trims off reductant features. The selection process is empowered by swarm search which uses a number of search agents to scout for a suitable subset of features that give rise to the highest possible accuracy. On the other facets, the data rows are filtered by using a misclassified filter which purges data instances preemptively before they enter further into the main predictive model building process. During the data cleaning process, a white learner is constructed because it is relatively faster and simpler to train a white-learner which could be used as a base learner in the classifier-based feature selection and misclassified removal. By the time when the data is cleansed, a white-learner is readily built to close perfection in terms of performance. The same data that was polished and used in building white-learner is sent to build a black-learner in the hope of achieving the highest accuracy. There are of course other alternatives in choosing the sequence of building models, cleaning data and passing data. This recommended approach should be one that saves time because a white-learner would have been trained anyway during the data pre-processing step.

The intellectual contribution from the works in this paper is three-fold: First, a WL methodology (passing cleaned data) is designed; second, two add-on's for swarm feature selection are applied: Early Abandonment, and Accelerated Swarm; thirdly, a qualitative analysis on Bayes Net from the white-learner by comparing with medical facts.

III. EXPERIMENT

The experiment is conducted over two datasets and has two respective purposes. First, we compare how medical datasets can be used to induce a model using three different types of machine learning algorithms –NBU, DL, and the WL. The second objective is to try reducing dimensions by applying metaheuristic search to do feature selection on two medical datasets; one is about recognizing breast cancer cells in numeric values, the other type is nominal data which records the symptoms and events of a disease, whether they would be leading to liver disorder.

A. Noise test

To address the challenge of fast and accurate binary-class classification from operation in a distributed AI medical environment, a noise-test is arranged which simulate small levels of noise in the data for representing an imperfect data transmitting environments. The testing data is simulated by adding artificially generated random noise, with increment of 1% all the way up to 20%. The objective is to observe how the loosely coupled white learning model perform in the presence of noise, which is not uncommon in distributed online information platform, running over affordable connectionless communication protocols.

B. Swarm Search and Early abandonment

In the following experiments, we aim at finding the most suitable swarm search algorithm to do feature selection that helps increase WL performance in data stream mining environment. In this section, we used two datasets, one is Breast cancer dataset, and the other is Liver Disorder dataset, which all include 1,000 instances and 10,000 instances respectively. We first run the original datasets in Massive Online Analysis benchmarking platform which simulates incremental learning in data stream mining environment, and set the sliding window size as 50, which means in one time the model will allow 50 data come in and get tested, trained and get performance. When we finished this step, we could get three sets of performance results, which are from WL, DL and NBU through data stream mining. Then it could tell us, under data stream mining, which algorithm have best performance. The next step is try to optimize the performance of WL. In the following experiments, we tried many swarm search algorithms to do feature selection. We used Ant, Bat, Bee, Cuckoo, Firefly, Flower, Genetic Algorithm, Harmony, Particle Swarm Optimization, and Wolf to join this competition.

As we known in the former experiments, we try to use swarm algorithms to improve White Learning performance. We tried 10 different ways. When we only after running out all the algorithms, we could get the best performance swarm, which takes time because have to run all the swarm methods throughout the whole load of data. Therefore, we consider whether we could have some early knowledge to know which swarm algorithm is quite suitable to this dataset. In this experiment, we also used two datasets, one is Breast cancer, and the other one is Liver disorder. The sampled dataset is obtained by running some swarm feature selection over the original dataset. Sampling over the original dataset is by Weka Reservoir Sampling function, which means the function could generate subsample by randomly picking certain percentage of data from original dataset, while, the classification ratio of the subset is the same as the classification ratio of the original data set. Therefore, we generate sub sample data set for each of the original dataset after swarm FS by 5 % increment from 5% to 95%. Then we run each sub dataset in WL. So we could get a beautiful chart. In this figure, the horizontal axis is from 5% to 100%, and the vertical axis is the accuracy. There are ten curves, representing ten different swarm algorithms. So, it could tell us useful information. The motivation for this experiment is to test the “minimum little” of sample dataset that is needed for providing satisfactory performance by swarm algorithm.

C. Accelerated swarm

We aim at accelerating those swarm algorithms. Making it lower time cost and get same or better accuracy. Instead of letting the agents in the swarm algorithms to start from random positions initially, we assign some “good” starting positions for the agents. The good starting positions are hinted from the quality of the feature (attribute) candidates. So that we would know in advance which features should be searched firstly that would potentially give better performance. One fast way to know the quality of the features is to do a quick statistics check. In our implementation we chose CV that stands for coefficient of variation. To compute CV value for each feature, one needs to only know the mean and standard deviation, which are quite fast and simple. It is believed that CV has relation to feature selection. It was believed that CV has a direct proportion to goodness of feature selection task. If a feature has a very small or almost zero CV values, that means the values of the feature do not vary at all or vary very little. In this case, there won’t be good results by including such feature in the candidate subset and do feature selection. In contrast, if a feature has a good CV value, the feature contributes to the effectiveness of feature selection [20]. Using this approach could make the whole swarm search converge earlier and more quickly. Compared to random start, the results are better and obtained earlier. That is why it is named as accelerated swarm search. Taken from the article which is written by the author and published, this diagram shows how CV feature selection works. For more details, readers are referred to [20].

E. Datasets

Two datasets from medical domain are used in the experiments. The first dataset has all numeric attributes, the other dataset has all nominal attributes. The first training dataset contain empirical mammogram data that come from a database of patients in a hospital. The data are anonymized. The dataset is consisted of 1000 records of patient’s cells information – there are 6 cells being circled by experienced oncologist, which are considered to be the most significant features that characterise the levels of the alleged disease. The attributes which are used to characterise the cells include the shape of the cells such as radius, perimeter, surface area, and the surface of the cell such as smoothness, compactness, concavity, concave points, symmetry, and fractal dimension of the cell. Total there are 60 attributes. The dataset comes from a PACS of electronic health records at the first affiliated hospital of Guangzhou University of Traditional Chinese Medicine, Guangzhou city, China. The sensitive fields of the are is anonymized for the confidentiality of patients. The DDSM and its curated breast imaging subset are the collaborative efforts between Massachusetts General Hospital, Sandia National Laboratories and the University of South Florida Computer Science and Engineering Department. The bright and scattered distribution of irregular calcification tissue in a breast MRI (illustrated in the blue circle in Fig. 7) is an important clinical symptom for the doctor to diagnose with breast cancer. This lesion can also be detected by the margin or border of normal tissue and sick area using their significant difference of obscured, circumscribed and speculated texture properties. Similar to left ventricle size prediction, it is also challenging to identify breast cancer very precisely due to the dense breast tissue under mammogram screening and the analogous symptoms from an infection or other breast diseases such as mastitis. The second dataset came from a HEPAR project that was conducted in the Institute of Biocybernetics and Biomedical Engineering of the Polish Academy of Sciences in co-operation with physicians at the Medical Centre of Postgraduate

Education. The HEPAR system contains a database of patient records of the Gastroenterological Clinic of the Institute of Food and Feeding in Warsaw. The data has 70 nominal attributes and 10,000 patient records. The presence or absence of the symptoms in the 70 attributes lead to whether liver disorder or not.

F. Performance Evaluation

The experiment follows a prequential evaluation scheme which is also called test-then-train strategy. First, the model is tested with each newly arrived instance. If there is enough test data to imply that new rules need to be generated, the decision tree or decision table will expand and the trained model will be updated. In our experiment, the sliding window will work. We set the window size to 50, each time there will be 50 data from the dataset pass the model. The classification performance will be evaluated from the start of the process... The prequential operations for a standard data stream mining environment and the prequential benchmarking environment for evaluating WL model are shown in Figures 4a and 4b respectively.

Since the real-time performance is a big issue in online cancer detection scenarios, the accuracy, kappa, and time cost criteria are used to evaluate the performance of data mining and data stream mining in this experiment. Accuracy is how many percent of data being correctly classified. Cohen's kappa coefficient (κ) is a statistic value which estimates the inter-rater agreement for qualitative data objects. It is usually considered to be a more convincible measurement value than accuracy of classification in data mining. Kappa is therefore regarded to be a measure of reliability for a data mining model.

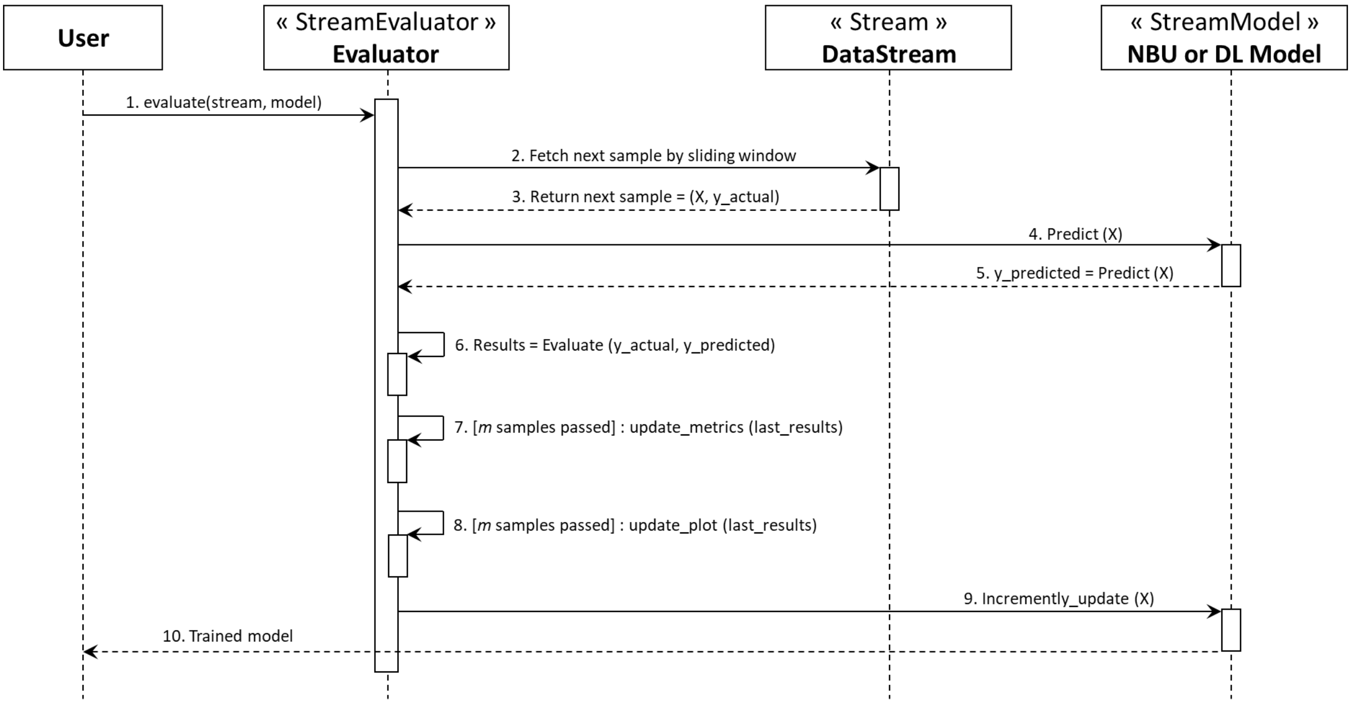

User
« StreamEvaluator »
Evaluator
« Stream »
DataStream
« StreamModel »
NBU or DL Model
1. evaluate(stream, model)
2. Fetch next sample by sliding window
3. Return next sample = (X, y_actual)
4. Predict (X)
5. y_predicted = Predict (X)
6. Results = Evaluate (y_actual, y_predicted)
7. [m samples passed] : update_metrics (last_results)
8. [m samples passed] : update_plot (last_results)
9. Incremently_update (X)
10. Trained model

Fig. 4a. Prequential evaluation for a typical data stream mining scenario in sequence diagram.

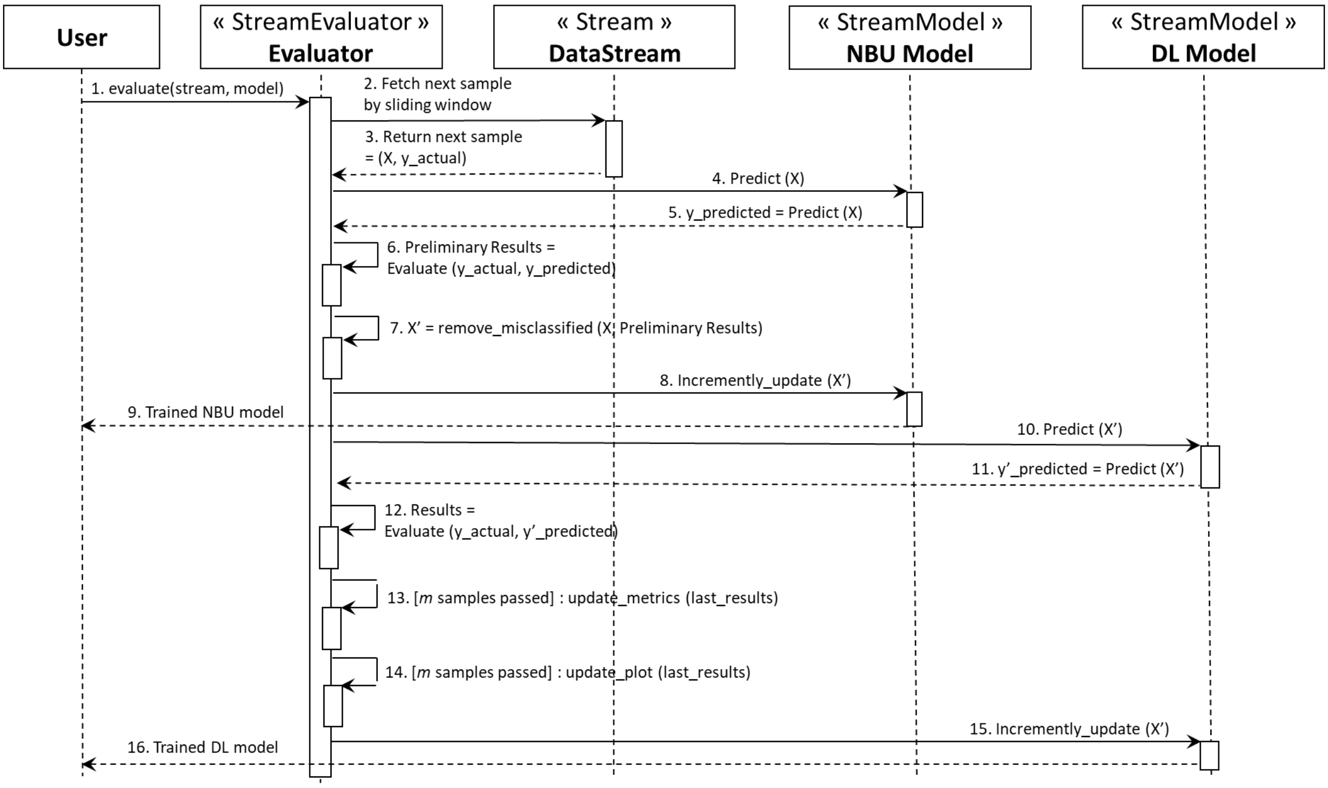

User
« StreamEvaluator »
Evaluator
« Stream »
DataStream
« StreamModel »
NBU Model
« StreamModel »
DL Model
1. evaluate(stream, model)
2. Fetch next sample
by sliding window
3. Return next sample
= (X, y_actual)
4. Predict (X)
5. y_predicted = Predict (X)
6. Preliminary Results =
Evaluate (y_actual, y_predicted)
7. X' = remove_misclassified (X, Preliminary Results)
8. Incremently_update (X')
9. Trained NBU model
10. Predict (X')
11. y'_predicted = Predict (X')
12. Results =
Evaluate (y_actual, y'_predicted)
13. [m samples passed] : update_metrics (last_results)
14. [m samples passed] : update_plot (last_results)
15. Incremently_update (X')
16. Trained DL model

Fig. 4b. Prequential evaluation for the proposed WL incremental learning model scenario in sequence diagram.

IV. RESULTS AND DISCUSSION

A. Noise test

The levels of accuracy, kappa and model construction time of White Learning, *NaiveBayesUpdatable* and Deep Learning are compared, and the results are shown in Figures 5.1.1-5.1.3.

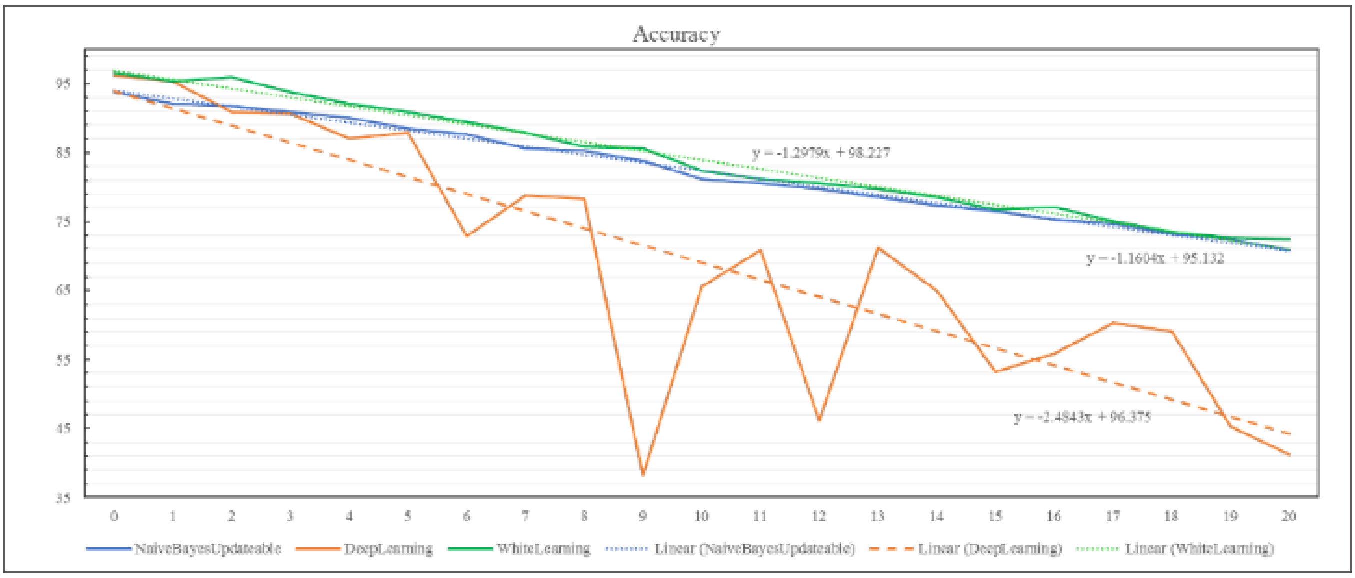


Fig. 5.1.1. The accuracy performance by using White Learning, NaiveBayesUpdatable and Deep Learning in cancer classification.

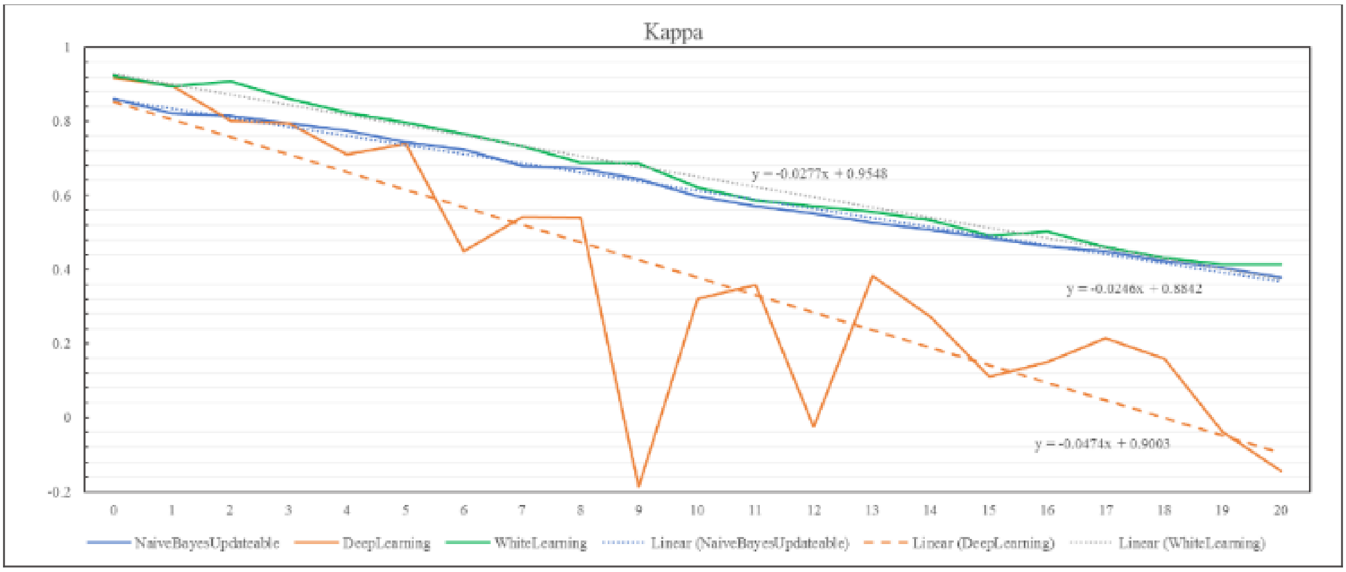


Fig. 5.1.2. The kappa performance by using White Learning, *NaiveBayesUpdatable* and Deep Learning in cancer classification.

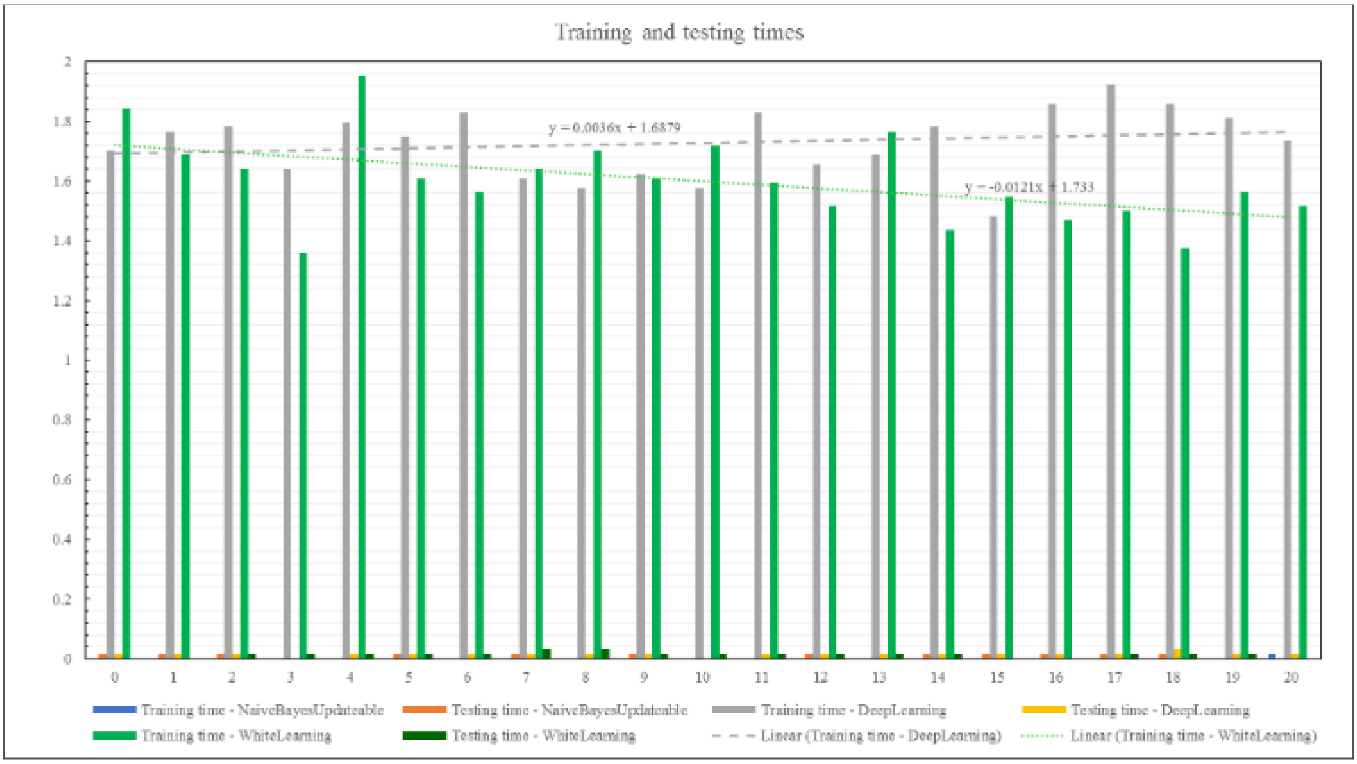


Fig. 5.1.3. The time performance by using White Learning, *NaiveBayesUpdatable* and Deep Learning in cancer classification.

The performance curves for accuracy and kappa are almost identical indicating that a good model is both accurate and generalizable for different new datasets and vice-versa. In Figure 5.1, it is easy to observe that the orange curve which is represent the Deep Learning method fluctuates greatly than the other two methods, White learning and *NaiveBayesUpdatable*. That shows DL is very sensitive to environment changing, when there is no noise being added, the performance of DL could reach 96% of accuracy, which is higher than that of NBU and almost equal to that of WL. As the noise keeping increasing, the DL start unable to control the situation and shows big drop at the noise level 6%. During the experiment, the accuracy of WL and NBU also decreases but smoothly and slowly in a linear way with increasing noise. In the worst case of adding 20% noise, the performances of WL and NBU are around 75% of accuracy and 0.5 of Kappa value, but the performance of DL is lower than that of random guessing. Among the WL and NBU, it is still clear to see that WL performance is slightly better than that of NBU at the most of time. In this experiment, WL is no doubt the best and the worst is DL. It is the best way to observe the differences between the 3 algorithms performance by setting the trend lines to each curve. The gradients of the curves which are represent the accuracy of Deep Learning, *NaiveBayesUpdatable* and white learning are -2.4843, -1.1614 and -1.2979 respectively That means within the same unit, the larger the absolute value of the gradient, the more intense the change. If the dataset is ideally pure without any noise, the accuracy levels for deep learning, *NaiveBayesUpdatable* and white learning are 96.1765%, 93.8235% and 96.4706% respectively. Even during the worst scenarios in the simulation, the accuracy levels are maintained around 41.1765%, 70.8824% and 72.3529% respectively. A high efficacy is demonstrated by white learning which is a hybrid comparing to NB and DL alone.

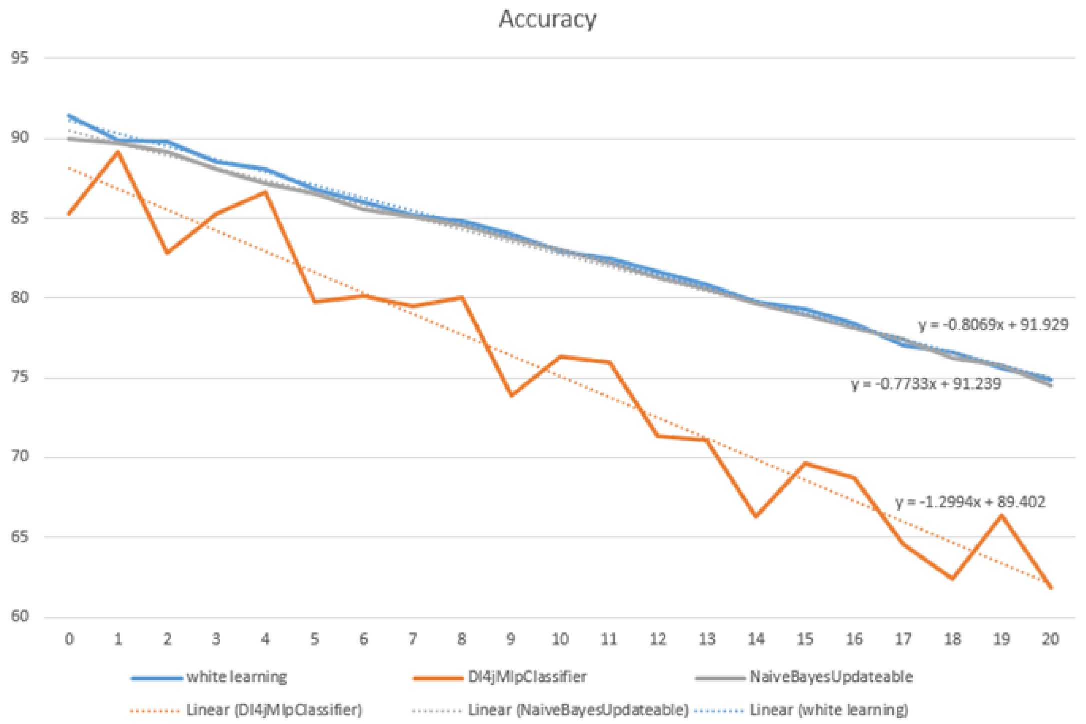


Fig. 5.1.4. The accuracy performance by using White Learning, *NaiveBayesUpdatable* and Deep Learning in liver disorder classification.

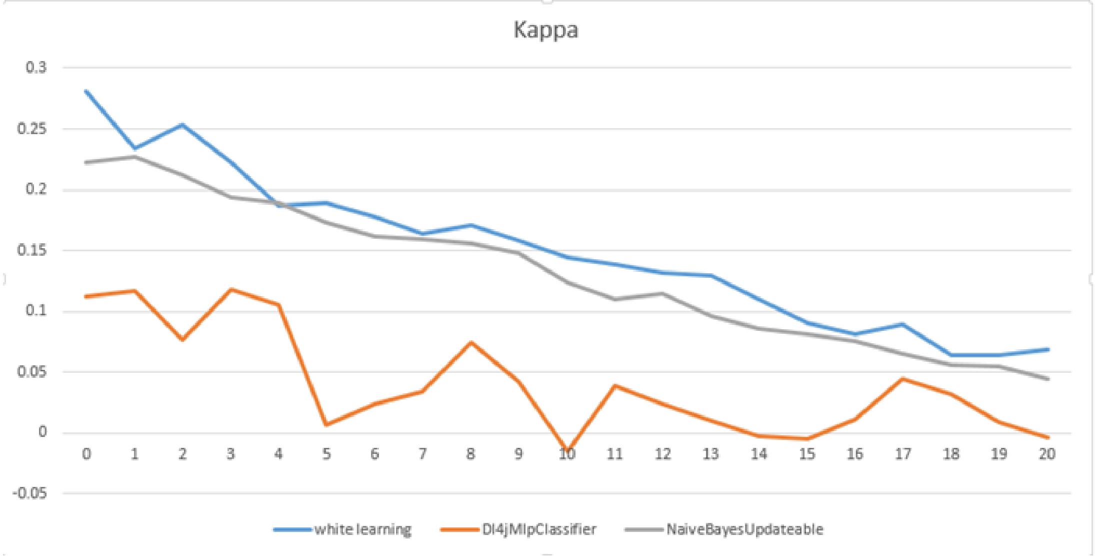


Fig. 5.1.5. The kappa performance by usingWhite Learning, *NaiveBayesUpdatable* and Deep Learning in liver disorder classification.

For Liver disorder dataset, it shows identical result trend with Cancer data result in both Accuracy and Kappa value. In Figure 5.1.4, it shows White Learning curve is always higher than curves of NBU and DL in accuracy. The accuracy result of Deep Learning also fluctuate a lot when noise added which is the same as the trend in Breast cancer dataset. It is the best way to observe the differences between the 3 algorithms performance by setting the trend lines to each curve. It can be observed that the gradients of the trend-lines for deep learning, *NaiveBayesUpdatable* and white learning are -1.299, -0.7733 and -0.807 respectively. If there is no noise added, all the performances could be around 90%. When noise added, the accuracy of DL rapidly decreases with a slop of -1.299 which is much higher than the slop of WL -0.807. That means within the same unit, the larger the absolute value of the gradient, the more intense the change. WL is the most sustainable under noise in terms of the rate of accuracy degradation. If the dataset is ideally pure without any noise, the accuracy rates for deep learning, *NaiveBayesUpdatable* and white learning are 85.2396%, 89.9735% and 91.4142% respectively. During the worst scenarios in the simulation, the accuracy levels are still sustained at 61.8935%, 74.4780% and 74.8603% respectively. Again, the efficacy of white learning is demonstrated comparing to NB and DL alone.

B. Swarm Search and Early abandonment

In the former experiments we test WL noise tolerance and scalability compared with WL and NBU. We found that WL has excellent performance in both aspects. In this experiment we try to use swarm algorithms to do feature selection to achieve two missions. Firstly, we try to improve WL performance in both accuracy and kappa value. Secondly, we try to find the most suitable swarm algorithm from more than 10 of them. In this experiment, we Massive Online Analysis (MOA) [21] is applied here due to its popularity in data stream mining, many researchers have used MOA as a benchmarking platform for evaluating data stream mining algorithms. In MOA we will set sampling frequency and member check frequency to 50. That means the system will give us a performance evaluation report after train-and-test per 50 instances. In this experiment, we also used 2 dataset which are Cancer dataset and Liver dataset. We firstly run the original dataset in MOA to compare WL with DL and NBU. Then we apply 10 different swarm algorithms to do feature selection to two original datasets respectively. We use datasets that have been pre-processed by swarm feature selection to test WL in MOA and record each performance. Finally, we can compare which algorithm would be the best.

In this experiment, we used accuracy and kappa as indicators to measure the quality of the model. For original cancer dataset, the accuracy, Kappa, Memory cost and Time results for comparing WL, DL and NB in MOA are shown in from Figure 5.3.1 to Figure 5.3.4 respectively. On the x-axis the variable is the number of instances, and it varies from 50 to 10,000 with an increment of 50 in each step. In Figure 5.2.1, it is observed that there is a big drop in the middle of the blue line, which represents the accuracy of NB in original Cancer dataset. At the same time, we can see from the Figure 5.2.2 that Kappa value of NB, which corresponds to the accuracy, also experienced a big drop. While, the other two lines maintain relatively high performance in both accuracy and Kappa value. We can see WL and DL are more stable and will not fluctuate a lot like NB. When we look at the front part of Figure 5.2.1, before the 2150 instances, the performance of both accuracy and Kappa value of WL is so much better than DL. When the abscissa value is around 1000 instance, both curve of accuracy of WL and DL reach the peak. But the accuracy of WL is 8% higher than DL, which are 80% and 72% respectively. Overall, WL is better than DL and NB in both accuracy and Kappa.

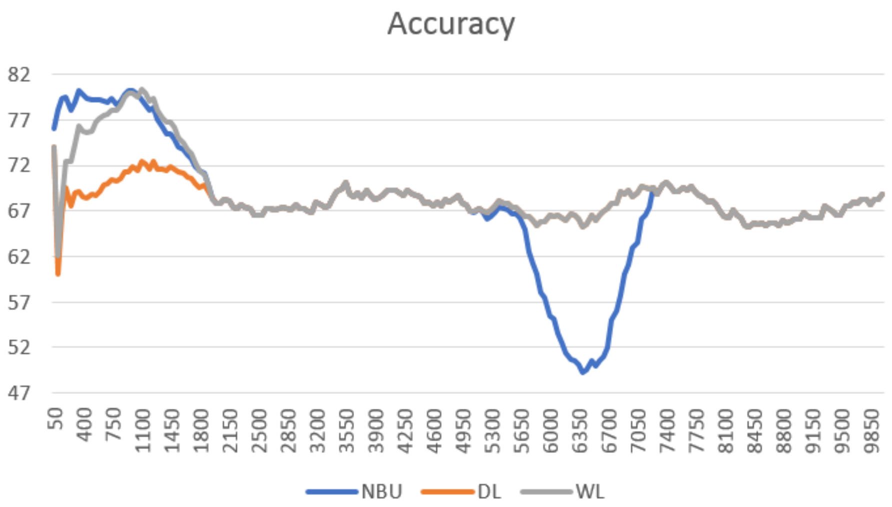


Fig. 5.2.1. The accuracy performance by using White Learning, *NaiveBayesUpdatable* and Deep Learning in cancer classification in MOA.

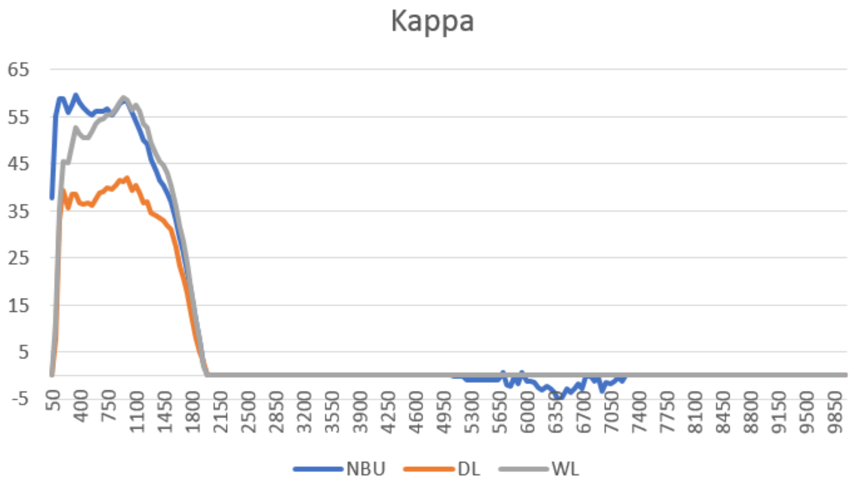


Fig. 5.2.2. The kappa performance by using White Learning, *NaiveBayesUpdatable* and Deep Learning in cancer classification in MOA.

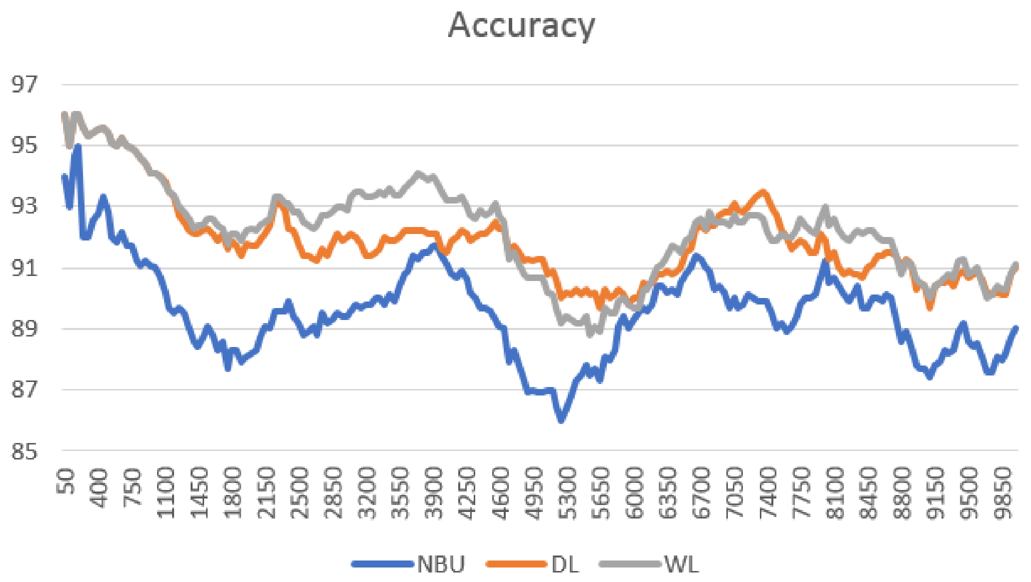


Fig. 5.2.3. The accuracy performance by using White Learning, *NaiveBayesUpdatable* and Deep Learning in liver disorder classification in MOA.

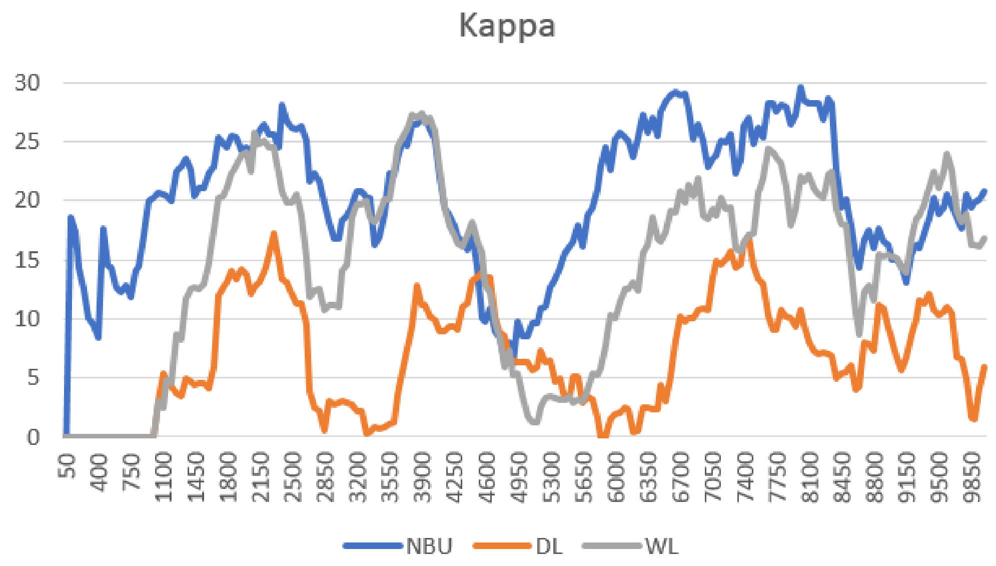


Fig. 5.2.4. The kappa performance by using White Learning, *NaiveBayesUpdatable* and Deep Learning in liver disorder classification in MOA.

For Liver dataset, the same trend of accuracy occurred. In the Figure 5.2.3, we can see the fluctuation of NB curve is much larger than the other two curves and it is always under the other two curves. That means, NB doesn't perform well in this experiment. As shown by the accuracy results, it can be observed that WL is better than DL in most cases. The Kappa result of WL is also in the middle of the other two curves.

In conclusion, WL is better than DL and NB in both accuracy and Kappa in this test.

B. Swarm Search and Early abandonment

After verifying the good performance of WL we try to improve it by using Swarm Search to do feature selection and pick the best swarm search algorithm. There are 11 swarm search algorithms on our testing list. They are Ant, Bat, Bee, Cuckoo, Elephant, Firefly, Flower, Genetic Algorithm, Harmony, Particle search optimization and Wolf. We also test these in both Cancer dataset and Liver dataset.

For the Cancer dataset, it is observed in Figure 5.3.1 and Figure 5.3.2, the average accuracy and kappa values are shown clearly in the bar chart. For convenience observation, we manually set a reference line according to the original data result. Then we can observe that, there are two swarm algorithms over the original one in both accuracy and kappa values which are Ant and Wolf. Especially, for the Wolf algorithm, its accuracy is higher than the original by 0.15% and Kappa value higher than the original by 0.1, which is a significant improvement.

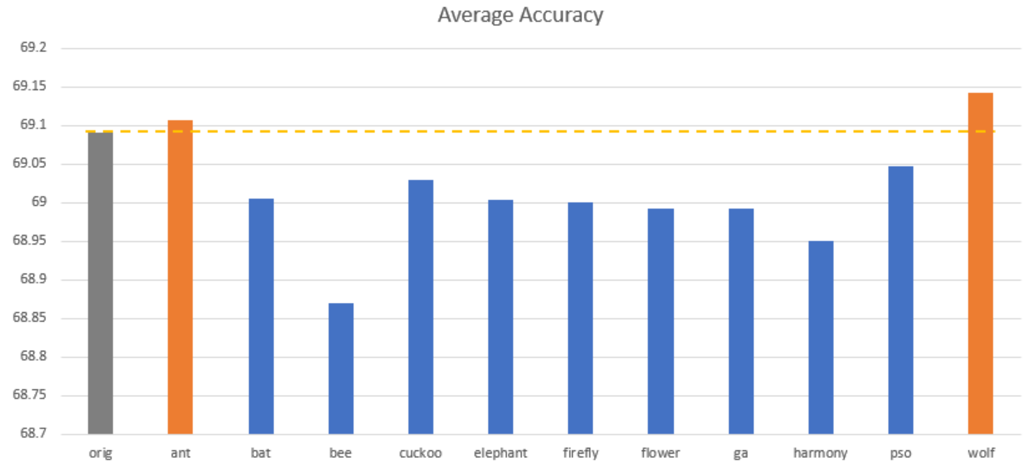


Fig. 5.3.1. Comparison of 11 kinds of swarm search algorithms applied in White Learning in terms of average accuracy for cancer dataset.

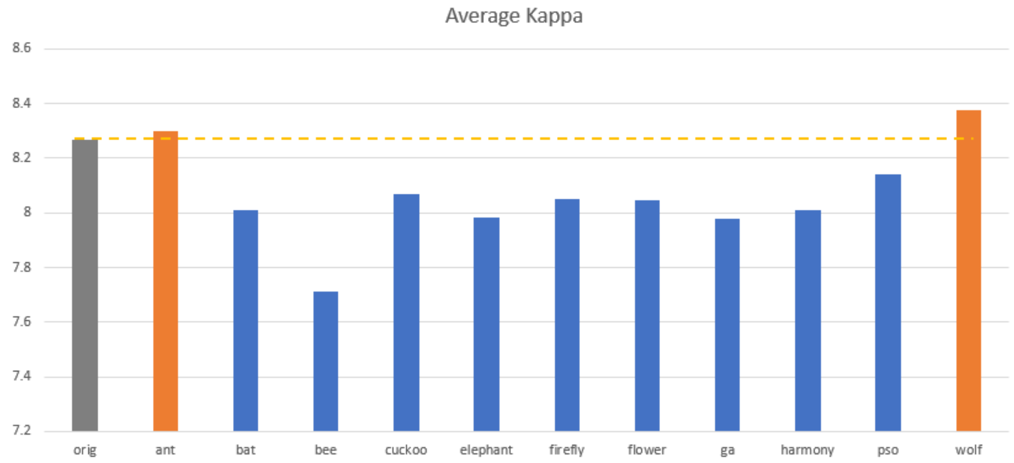


Fig. 5.3.2. Comparison of 11 kinds of swarm search algorithms applied in White Learning in terms of average kappa for cancer dataset.

For Liver disorder dataset, we can see that Figure 5.3.3 and Figure 5.3.4, which is show the average accuracy and average Kappa values respectively. It is shown in Figure 5.3.3, there are 6 kinds of swarm search accuracy result higher than the original one. Among them, Flower algorithm own the best average accuracy, which is 1.5% higher than the original dataset result. For the average Kappa value, we can observe that there are also 6 swarm algorithms result higher than the original result. But interestingly, the average kappa of flower is near 0. That means, although Flower could achieve higher accuracy, it is not stable. So, at the end of the experiment, we find that there is only one swarm algorithm, both its average accuracy and its average kappa value are higher than the original one, which is Wolf swarm search algorithm.

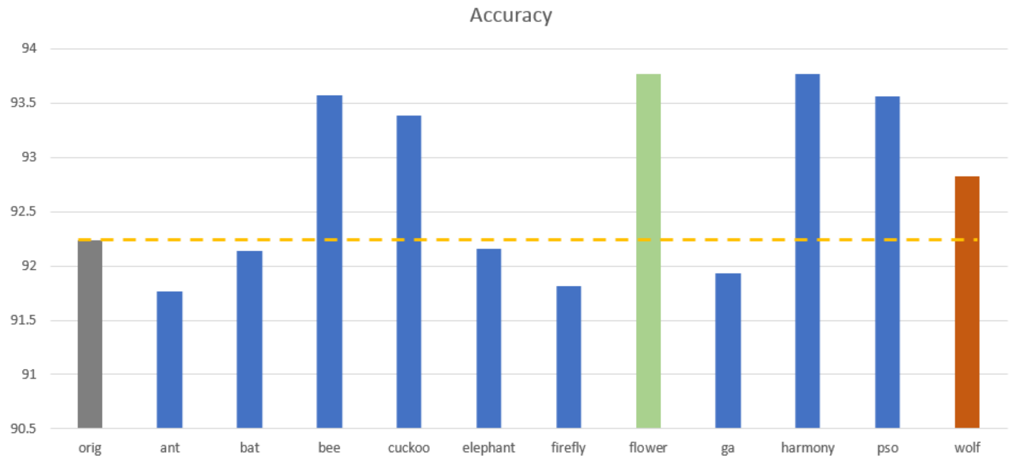


Fig. 5.3.3. Comparison of 11 kinds of swarm search algorithms applied in White Learning in terms of average accuracy for cancer dataset.

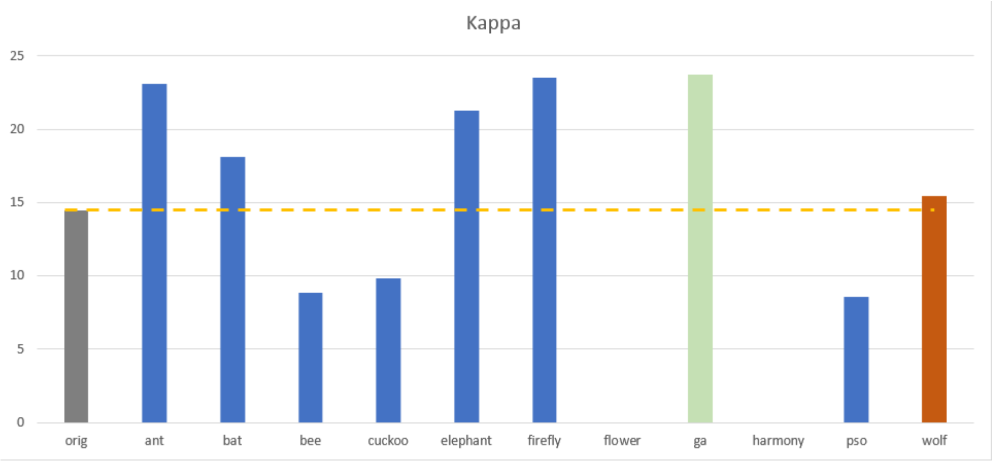


Fig. 5.3.4. Comparison of 11 kinds of swarm search algorithms applied in White Learning in terms of average kappa value for cancer dataset.

From the experiments above, we applied more than 10 kinds of swarm search algorithms to improve the WL performance. After testing all the algorithms, we will know which algorithm work best. Running the whole length of datasets which could be very large in big data environment, takes time, and very cumbersome. Only after running all the experiments, we could know which swarm algorithms are not suitable, and should be abandoned. What if we could know it earlier before we run for the full lengths of data in the experiment? So, we designed this experiment to choose the best swarm search algorithm, using data sampling approach. This approach could be applied at model calibration where small samples of datasets are used to test the efficacy of swarm search methods.

In this experiment, our goal is to early know which swarm search algorithms can be retained and which should be early terminated without training all the data in the dataset. Therefore, we first divide all datasets into subsets of 5% to 95% with an increase rate of 5%. The method applied is by using WEKA *ReservoirSample* function. It generates a random subsample by applying the reservoir sampling algorithm. Each subset maintains the same classification ratio as the original dataset or after running different swarm search algorithms datasets.

In this experiment, we only test Liver dataset because it is 10 times larger than cancer and it has 10 attributes more than the cancer dataset. So, for the sake of testing the efficacy of sampling for early abandoned, it is preferred to use a larger dataset. It is anticipated that similar results would be generated for the cancer dataset, therefore the test is not repeated for the cancer dataset.

For the Liver dataset, as we have known earlier, all the swarm search algorithms improved the WL performance. The top 5 from the previous test are PSO, Harmony, Cuckoo, Bee, Wolf. According to our experimental theory, the swarm search algorithms that performed well in the former experiment should show the same advantages in this experiment. For the convenience of observation, we artificially set three peak lines in Figure 5.3.5. It can be observed that almost every swarm search algorithm get the peak and the values almost have no difference. Except we can say PSO get the Top position in the first peak. At the second peak, PSO, Bee, Cuckoo, Wolf, Ant and Harmony are ranked high. At the third peak, Cuckoo, PSO, Wolf, Bee, Elephant, Harmony and Bat have a good performance. The rankings of these three peaks are roughly consistent with our ranking in the above experiment. In other words, those perform well swarm algorithms we should remain them run out all the dataset. While, those performing bad, like Firefly which always at the bottom and Wolf which fluctuated a lot and can be seen in the top peak and deep bottom, should be abandon early.

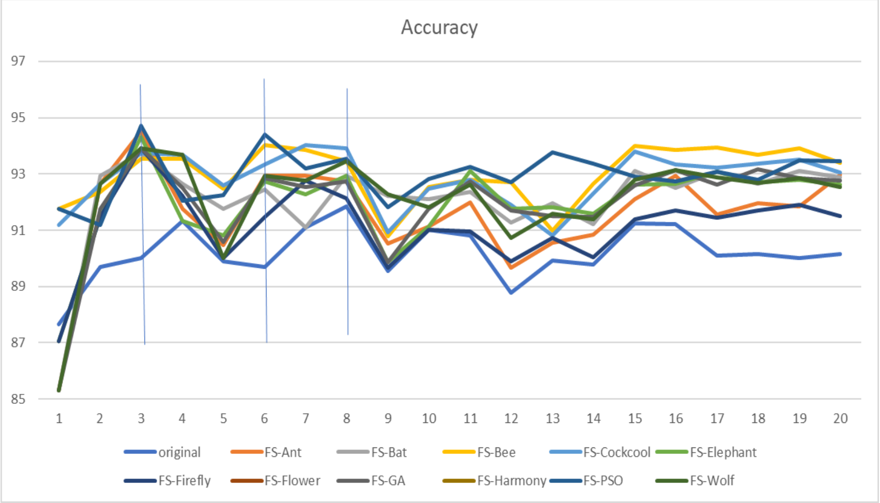


Fig. 5.3.5. Comparison of 11 kinds of swarm search algorithms applied in White Learning in terms of average accuracy for Liver dataset for early abandonment.

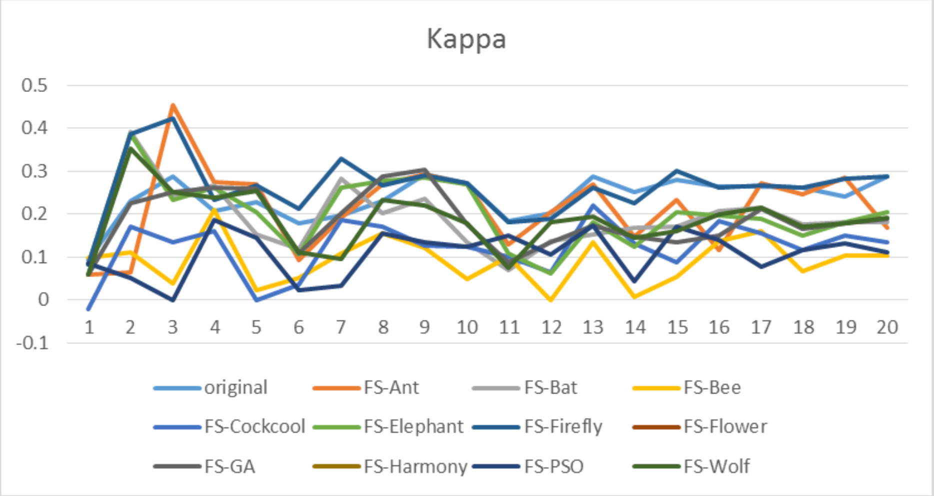


Fig. 5.3.6. Comparison of 11 kinds of swarm search algorithms applied in White Learning in terms of average kappa for Liver dataset for early abandonment.

C. Accelerated swarm

From the results charted in the Figures below, Accelerated swarm search methods have advantages in improving WL accuracy in general. The improvement is obvious except for Flower search and Bee search. For Kappa however, accelerated swarm search methods seem to under-perform compared to normal swarm search methods. For cases of PSO, Flower, Bee and Cuckoo, the Kappa values are about the same. The lower Kappa may be explained by that the accelerated search (that leads to early convergence) might not have covered many possibilities thereby compromising the generalization of the model hence lower Kappa value.

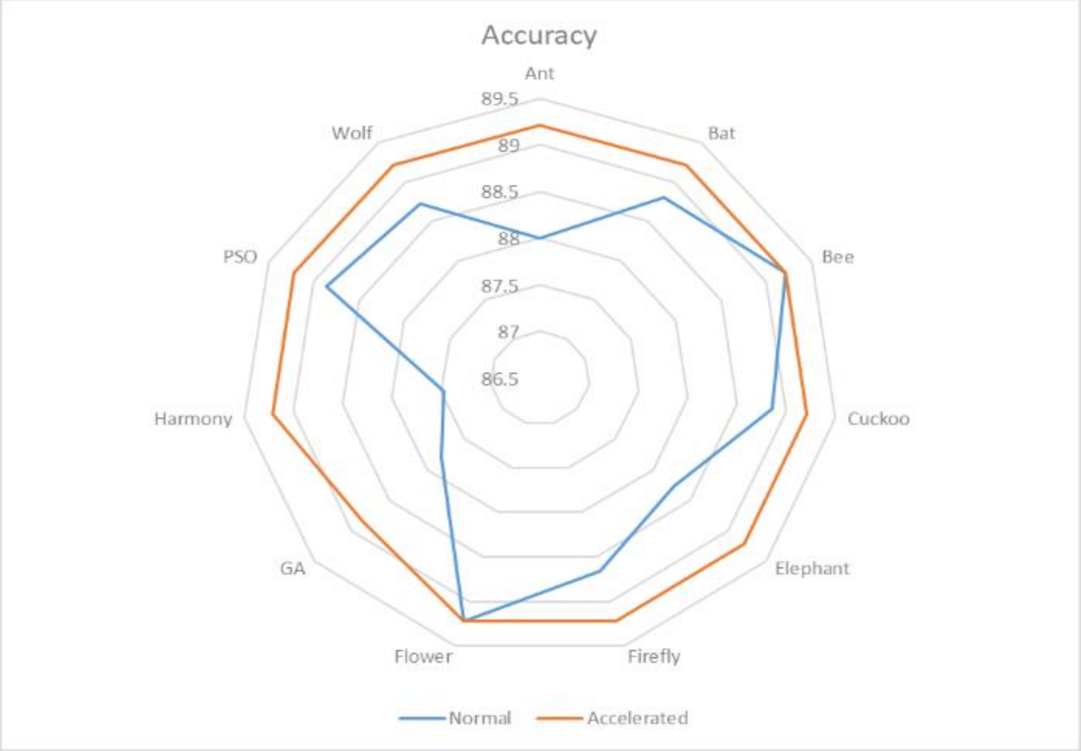


Fig. 5.4.1. Accuracy results of improved SSFS by Accelerated Swarm Search for WL.

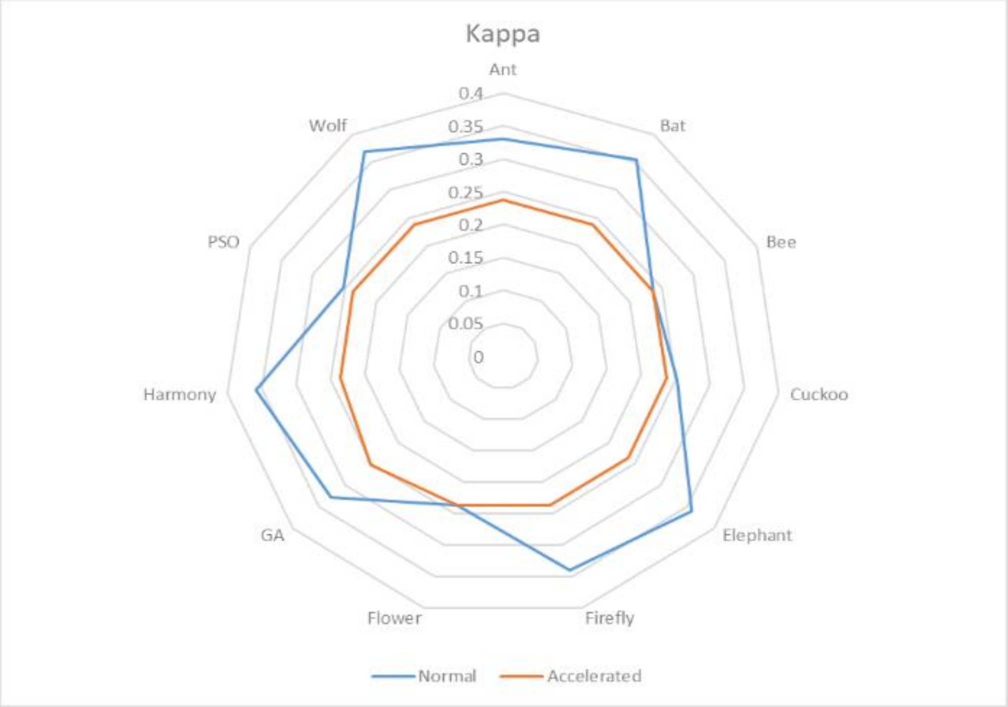


Fig. 5.4.2. Kappa results of improved SSFS by Accelerated Swarm Search for WL.

V. QUALITATIVE ANALYSIS OVER BAYESIAN NETWORKS

The qualitative analysis is planned to systemically showcase the Bayesian network as a result of using a collection of machine learning. It is started from the most basic Naïve Bayes learner over (1) the original liver disorder dataset, (2) the same dataset that has been applied with Misclassified Recall, and (3) with swarm feature selection methods. The objective is to observe the differences in Bayesian networks that are generated from different tools.

Figures 6.1 shows a full structure of Bayesian network. Since the full dataset is used in inducing this Bayesian network, the relations are at their fullest as well, having many links weaving through many nodes. Six possible causal paths link up factors leading to the destination node. Tracing the six paths, there are a collection of yellow nodes which serve as originators where the causal relations started from. Under examination of the yellow nodes, it is discovered that the originators nodes are quite general, for instances hospital, age, diabetes and gallstones etc. It is true enough that these attributes do have certain relations to carcinoma. Those source nodes might not be the original causes by common sense. The relationship between nodes (especially the relationship between initiators) is inaccurate, which may be due to poor base mapping between attributes and targets in the original dataset. Nevertheless, this preliminary demonstrates that it is technically possible to generate an interpretable Bayesian network for medical users to investigate the relations among factors and how they lead to a target.

In the next experiment, Misclassified Recall (MR) which is the core function in WL is applied on the liver order data. It is noticed that as shown in Figure 6.2, the structure of the Bayesian network is exactly the same as the one in Figure 6.1. By using MR, certain amount of data instances are removed because they cause confusion in the machine learning process. These misclassified data, once cleansed, the dataset is left with only good quality data which help induce a more accurate prediction model. In this case, the dataset is modified while the attributes remain the same. Therefore, there is no change in the Bayesian network structure. However, the links as shown in Figure 6.2 are different from the links in Figure 6.1. The links which represent the causal relations have changed totally after MR, indicating that new insights could be revealed from this new Bayesian network after MR.

Under a close observation, in Figures 6.2 and Figures 6.2 a-e, there are five paths existing in the Bayesian network. It indicates that with misclassified applied, liver cancer originates from something called AMA. What is AMA? In the following article, there is a mention of the relation between AMA and liver cancer:

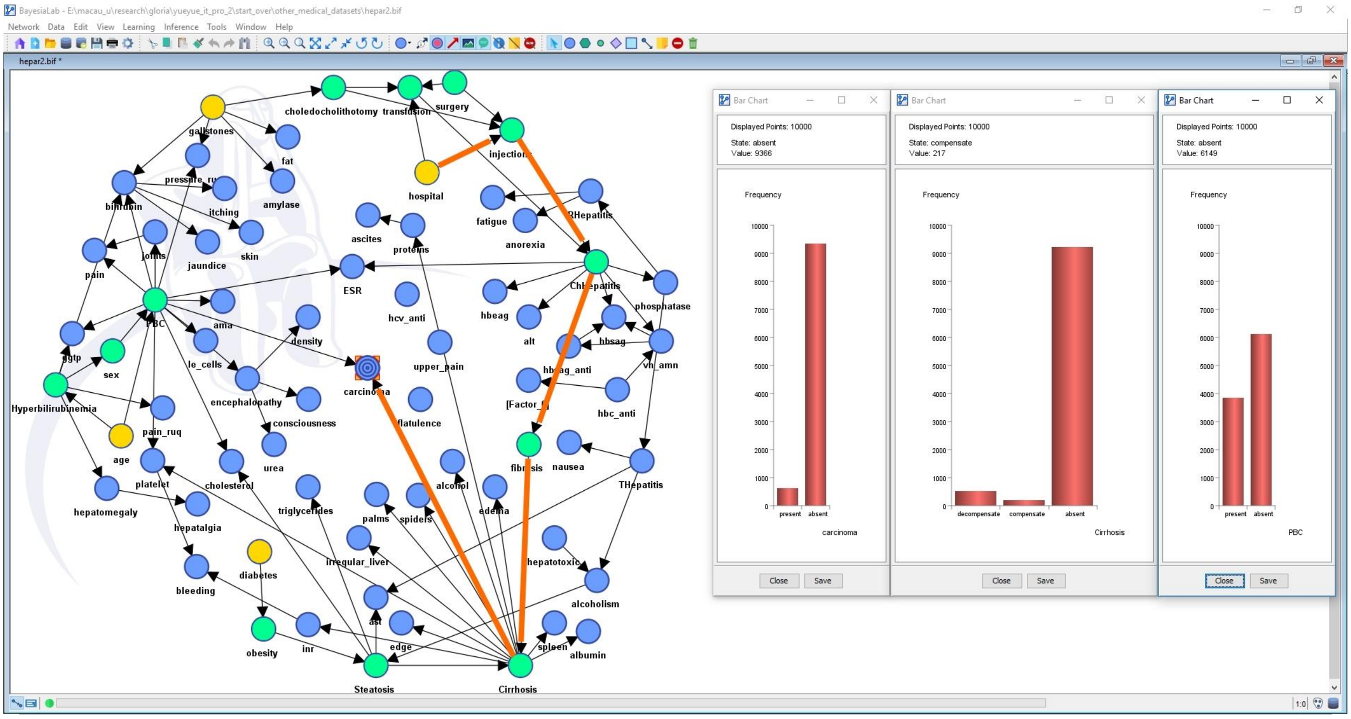
https://www.cancertherapyadvisor.com/home/decision-support-in-medicine/gastroenterology-hepatology/primary-biliary-cholangitis/

Most importantly, there is this sentence: "Approximately 95% of patients will test positive for antimitochondrial antibodies (AMA). AMAs are highly specific for PBC and are usually the first laboratory abnormality to occur." The casualty paths in Figure 6.2 indicates just this phenomenon. The yellow nodes which are the sources, originated from AMA, leading to PBC and other factors related to the disease.

In the Bayesian network by WL diagram, it shows AMA is highly related to PBC. Another medical article proves the same. Most importantly, there is this statement in the article: "Antimitochondrial antibodies (AMA) are autoantibodies that are strongly associated with primary biliary cholangitis (PBC), formerly called primary biliary cirrhosis. " https://labtestsonline.org/tests/antimitochondrial-antibody-and-ama-m2.

The conclusion is: one can see how closely consistent the Bayesian network by WL to the actual liver cancer information is as reported in medical documents! Essentially, it shows if we don't apply WL, we get some rather random results as in the first diagram by just using NB alone. After misclassified recall is applied, the WL has increased accuracy of the model via the cleansed data. It can also be noticed the diagram is clearer without many unnecessary nodes and links as compared to the earlier NB diagram in Figure 6.1.

Furthermore, after swarm feature selection is applied, the structure of the Bayesian network is simplified because many nodes are removed. The relations are shortened and became fewer too. However, the most important factors such as AMA and PBC are still preserved in the Bayesian networks which are resulted from using swarm feature selection and MR.



BayesiaLab - E:\macau_u\research\gloria\yueyue_it_pro_2\start_over\other_medical_datasets\hepar2.bif

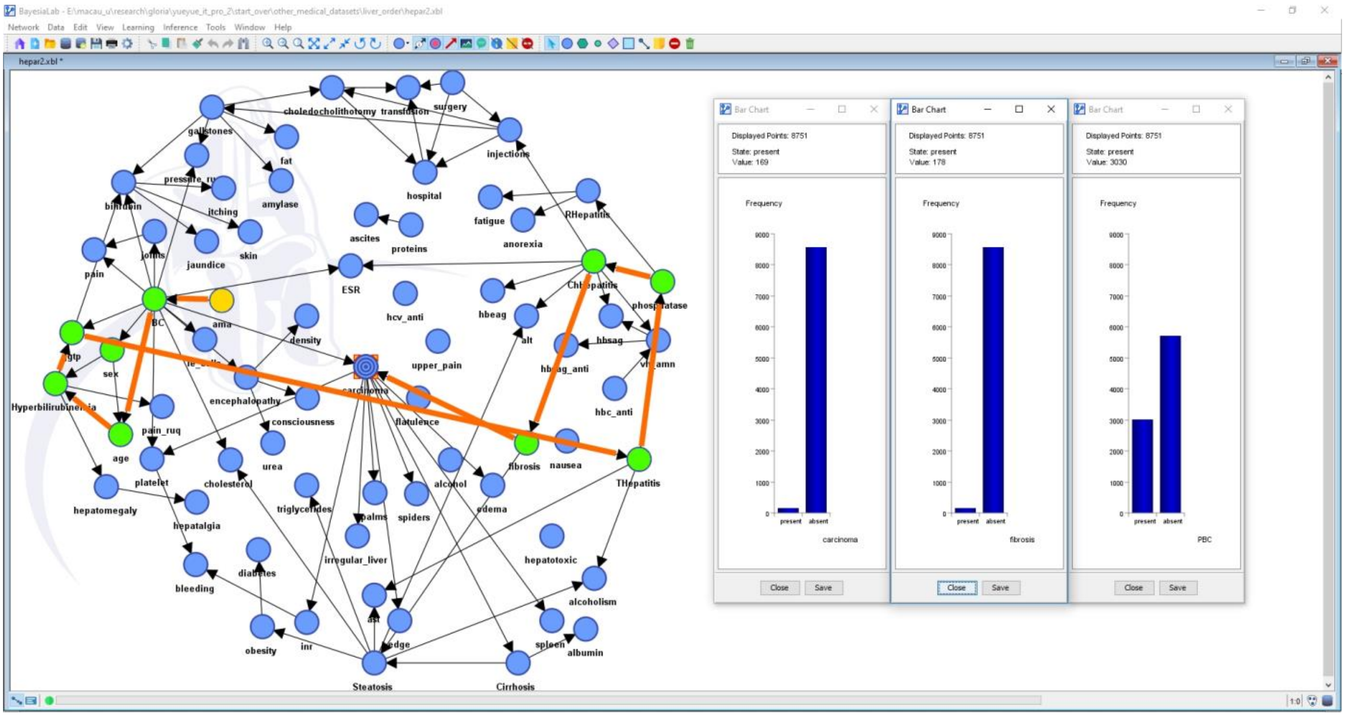
hepar2.bif *

Bar Chart — Displayed Points: 10000; State: absent; Value: 9366

Bar Chart — Displayed Points: 10000; State: compensate; Value: 217

Bar Chart — Displayed Points: 10000; State: absent; Value: 6149

Fig. 6.1. Bayesian network by original liver disorder dataset.



BayesiaLab - E:\macau_u\research\gloria\yueyue_it_pro_2\start_over\other_medical_datasets\liver_order\hepar2.xbl

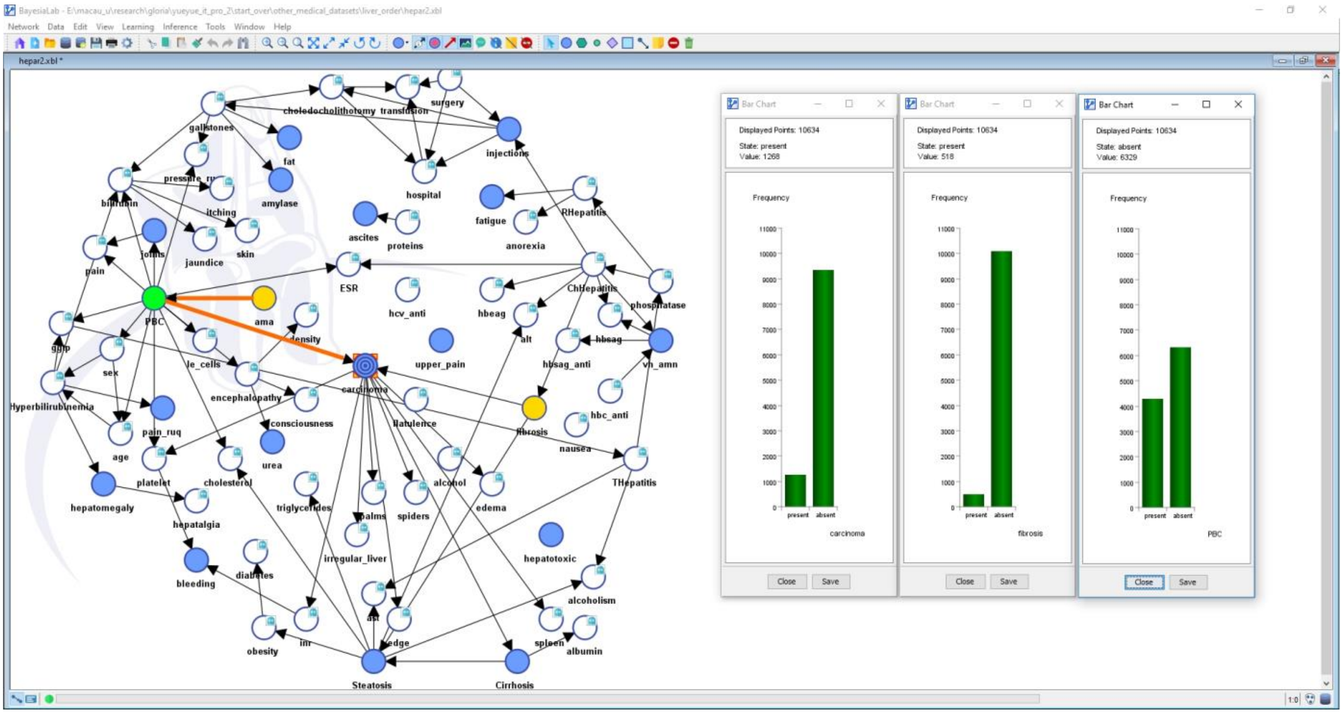
hepar2.xbl *

Bar Chart — Displayed Points: 8751; State: present; Value: 169

Bar Chart — Displayed Points: 8751; State: present; Value: 178

Bar Chart — Displayed Points: 8751; State: present; Value: 3030

Fig. 6.2. Bayesian network by misclassified recall liver disorder dataset.



Displayed Points: 10634
State: present
Value: 1268

Displayed Points: 10634
State: present
Value: 518

Displayed Points: 10634
State: absent
Value: 6329

Fig. 6.3. Bayesian network by misclassified recall + FS-WSA liver disorder dataset.

VI. Concluding Remarks

In general, as concluding remarks, if the user is in a hurry to extract only the most significant causality result from the data, swarm feature selection + MR should be used. It results in very concise and may be even overly simplified Bayesian network which shows only the strongest causality links. Otherwise, if fuller information in terms of more causal paths and factors are opted to be observed, the user can choose without feature selection. But it is still advisable to apply MR as in the WL model to retrieve paths that are very relevant to the prediction target. Otherwise, if just applying NB to generate a Bayesian network from the original dataset, the resultant causal paths may not be so accurate as the sources seem to be irrelevant to the disease.

Acknowledgement

The authors are grateful for the financial support from two the following research grants: (1) Nature-Inspired Computing and Metaheuristics Algorithms for Optimizing Data Mining Performance, Grant no. MYRG2016-00069-FST, by the University of Macau; and (2) A Scalable Data Stream Mining Methodology: Stream-based Holistic Analytics and Reasoning in Parallel, Grant no. FDCT/126/2014/A3, by FDCT Macau; and (3) 2018 Guangzhou Science and Technology Innovation and Development of Special Funds, Grant no. EF003/FST-FSJ/2019/GSTIC.

References

[1] Kunio Doi, Computer-Aided Diagnosis in Medical Imaging: Historical Review, Current Status and Future Potential, Comput Med Imaging Graph. 2007; 31(4-5): 198–211

[2] Kun Lan, Dan-Tong Wang, Simon Fong, Lian-Sheng Liu, Kelvin K. Wong, A Survey of Data Mining and Deep Learning in Bioinformatics, Journal of Medical Systems archive, Volume 42 Issue 8, August 2018, Pages 1-20

[3] Joao F. G. de Freitas, Bayesian Methods for Neural Networks, Cambridge University Engineering Department, PhD Thesis, 2009

[4] Wolfgang Fruehwirt et al., Bayesian deep neural networks for low-cost neurophysiological markers of Alzheimer's disease severity, NeurIPS 2018 (Conference on Neural Information Processing Systems), Machine Learning for Health Work-shop, Dec 2019, pp.1-6

[5] Raanan Y. Rohekar, Shami Nisimov, Yaniv Gurwicz, Guy Koren, Gal Novik, Constructing Deep Neural Networks by Bayesian Network Structure Learning, 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montr¨¦al, Canada, pp.1-12

[6] Iti Chaturvedi, Erik Cambria, Soujanya Poria, Rajiv Bajpai, Bayesian Deep Convolution Belief Networks for Subjectivity Detection, 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW), Barcelona, Spain, 12-15 Dec. 2016, pp.916-923

[7] Krakovna, Viktoriya. 2016. Building Interpretable Models: From Bayesian Networks to Neural Networks. Doctoral dissertation, Harvard University, Graduate School of Arts & Sciences.

[8] Jong Pill Choi1, Tae Hwa Han, Rae Woong Park, A Hybrid Bayesian Network Model for Predicting Breast Cancer Prognosis, Journal of Korean Society Medical Informatics 2009:15(1):49-57

[9] Peter Antal, Geert Fannes, Dirk Timmerman, Yves Moreau, Bart De Moor, Bayesian applications of belief networks and multilayer perceptrons for ovarian tumor classification with rejection, Artificial Intelligence in Medicine 29 (2003) 39–60

[10] Ashutosh Garg, Vladimir Pavlovic, Thomas S. Huang, Bayesian Networks as ensemble of Classifiers, Object recognition supported by user interaction for service robots, 11-15 Aug. 2002, 779-784

[11] Shuai Pang, Yuhan Jia, Rebecca Stones, Gang Wang, Xiaoguang Liu, A combined Bayesian network method for pre-dicting drive failure times from SMART attributes, 2016 International Joint Conference on Neural Networks (IJCNN), 24-29 July 2016, pp.4850-4856

[12] M. Correa, C. Bielza, J. Pamies-Teixeir, Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process, Expert Systems with Applications, Volume 36 Issue 3, April, 2009, Pages 7270-7279

[13] Tengyue Li, Simon Fong, Liansheng Liu, Xin-She Yang, Xingshi He, Jinan Fiaidhi, Sabah Mohammed, White Learning: A White-Box Data Fusion Machine Learning Framework for Extreme and Fast Automated Cancer Diagnosis. IT Professional 21(5): 71-77 (2019)

[14] John, G.H., Langley, P., Estimating Continuous Distributions in Bayesian Classifiers, In: Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence, pp.338-345. Morgan Kaufmann Publishers Inc. (1995)

[15] Novet, Jordan (2015-11-14). "Want an open-source deep learning framework? Take your pick". VentureBeat. Retrieved 2015-11-24.

[16] Simon Fong, Xin-She Yang, Suash Deb, Swarm Search for Feature Selection in Classification, 2013 IEEE 16th International Conference on Computational Science and Engineering, 3-5 Dec. 2013, pp.902-909

[17] Simon Fong, Jiaxue Li, Wei Song, Yifei Tian, Raymond K. Wong, Nilanjan Dey, Predicting unusual energy consumption events from smart home sensor network by data stream mining with misclassified recall. J. Ambient Intelligence and Humanized Computing 9(4): 1197-1221 (2018)

[18] Simon Fong, Shimin Hu, Wei Song, Kyungeun Cho, Raymond K. Wong, Sabah Mohammed, On Recognizing Abnormal Human Behaviours by Data Stream Mining with Misclassified Recalls. WWW (Companion Volume) 2017: 1129-1135

[19] Pedro Alves, Shuang Liu, Daifeng Wang, Mark Gerstein, Multiple-Swarm Ensembles: Improving the Predictive Power and Robustness of Predictive Models and Its Use in Computational Biology, IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) TCBB Homepage archive, Volume 15 Issue 3, May 2018, Pages 926-933

[20] Simon Fong, Justin Liang, Raymond Wong, Mojgan Ghanavati, A novel feature selection by clustering coefficients of variations, Ninth IEEE International Conference on Digital Information Management (ICDIM 2014), 29 Sept.-1 Oct. 2014, pp.205-213

[21] Albert Bifet, Geoff Holmes, Richard Kirkby, Bernhard Pfahringer (2010); MOA: Massive Online Analysis; Journal of Machine Learning Research 11: 1601-1604

Appendix A - Definition

White-learning model (WL). WL is an ensembled supervised machine learning model which consists of minimum one white-learner and one black-learner, so WL provides dual output results - one from the black-learn which usually is focused to achieve a prediction with the highest possible accuracy, and an explainable graph which can be interpreted by the users to know how the prediction is derived in the induction and/or deduction process. The two types of leaners are supposed to be coupled as a hybrid at various levels. (More details follow in the next sub-section.) In our case, the WL model we implemented and tested in experimentation is most basic one which connects a typical black learner and a classical white learner by passing data. The machine learners in WL need to operate in incremental manner, taking only one portion of incoming training data at a time. Inside WL model, there are optional pre-processing tools designed for enhancing the performance of WL by improving the quality of the training data. The black learner and white learner could be implemented by different algorithms, as long as they output dual prediction results, in the form of an accurate prediction and some patterns which could be understood by human users. In the context of experimentation and application as documented in this thesis, WL is generally referred to a methodology of employing both white-box learner and black-box learn to generate dual prediction results. WL consists of both black-box and white-box algorithms.

Black-learner (BLN). BLN is a computer program algorithm which induces training data into a learnt supervised machine learning model, without any knowledge of its internal workings; likewise, it outputs a result without explaining how it is inferred. The output is often just a univariate number or nominal answer. The performance of the prediction often could be evaluated.

White-learner. WLN is a computer program algorithm which induces training data into a learnt supervised machine earning model, with knowledge of its internal workings, such as probabilistic learning; likewise, it outputs a result with explanation about how it is inferred. The output is often dual that predicts a univariate number or nominal answer, as well as showing the relations between the predictor variables and the prediction target, e.g. decision trees, Bayesian network and classification-based association rules. The performance of the prediction often could be evaluated.