Machine Learning in a Changing World

Joel Scanlan, Jacky Hartnett and Ray Williams
School of Computing and Information Systems, University of Tasmania

Abstract

For many different fields of research the target of the research evolves over time. In this paper a machine learning approach called DynamicWEB is presented. This allows a knowledge hierarchy to evolve as the result of changes in multiple observations of the same entities. By adapting the hierarchy to these changes, the learning technique is able to preserve and improve itself instead of being affected negatively by it. Several machine learning datasets are examined that demonstrate the value of this technique.

1. Introduction

Machine learning and data mining are utilized in many different scientific and industrial domains. They are applied in an effort to extract knowledge or meaning from a vast array of different data sources. Advances that are made when the focus is on a particular problem space can often be successfully applied in other knowledge domains. Within this paper a method that was developed while examining one problem space, will be shown to be a useful machine learning technique that can be applied in other domains.

The fields of data mining and machine learning aim to extract patterns that are present within a dataset. Included in this is the process of examining datasets in which the patterns that are sought actually change over the course of the dataset. Change is not a particularly novel concept; it is something that could occur in every domain in which machine learning and data mining techniques is used. However, many techniques do not adapt well to change, making them unsuitable, for use on particular problems. Alternatively, they may require that the model be reformed periodically to account for the change that may have taken place.

Two kinds of changes may occur to the classifications found within a data set. Concept drift has been examined by multiple authors [1, 2] and is the phenomenon in which a description of a resulting class may change over the course of a dataset. Object drift [3] is where a object that has been examined multiple times may change its characteristics so that from being originally considered a member of one class it is later associated with another. In this paper, the conceptual clustering technique called DynamicWEB will be shown to provide for both of these conditions. This paper will present results based on several small machine learning datasets to illustrate this.

2. Concept and Object Drift

Many machine learning and data mining methods operate under the assumption that they are examining data that does not contain a natural order. As a result when various trials are repeatedly run, and the ordering of the data is randomized in between, the context contained within the dataset order is lost. However, there are methods that not only operate on sequential data but are also designed to investigate ordered data with the goal of utilizing change within the dataset to produce flexible models that changes over time.

Among the foundational methods that provide for *concept drift* within a data set are STAGGER [1] and FLORA [2]. These use supervised learning and both, as well as other methods that have built upon them, adapt to concept drift as it occurs, whether this is a gradual change over time, or a sudden dramatic change. This adaption allows for predictive accuracy to be recovered or maintained when concept drift has occurred. Without this, the predictive accuracy of the learner may suffer in the long term.

A related problem occurs when a target object within a particular data set is observed multiple times by a learner and over time one or more of the object's characteristics change. The learner then has to cater for this *object drift* by enabling a profile of this target object be updated and perhaps reclassified within the model that is being produced.

3. Conceptual Clustering

Conceptual Clustering was first described by Michalski [4] and then expanded with Stepp [5] who introduced the PAF method. Conceptual clustering aims to derive concept descriptions from the dataset. These descriptions allow for clusters to have a simple conceptual interpretation. In this way conceptual clustering extends data clustering methods, by not only discovering the relationships within the data, but also discovering human readable clusters. Furthermore, these classes fit descriptions which illustrate a true "is-a" relationship allowing for relationships to visibly extend downwards in the knowledge hierarchy produced. As each of the descriptions, or concepts, are formed they are placed in a tree structure, with the concepts that are broad towards the root, and more specific concepts nested within those higher parent concepts as children.

This paper presents a method, entitled DynamicWEB, which at its core is a substantial modification to the COBWEB unsupervised conceptual clustering algorithm. This is not the first time that the COBWEB algorithm has been modified by other researchers, and indeed other work has been completed to allow it to adapt to concept drift. This other work is titled COBBIT and will be also briefly explained.

COBWEB

The COBWEB algorithm was published by Fisher [5] and builds upon the work completed by Michalski in PAF [4], and the UNIMEM [6] and CYRUS [7] algorithms. While COBWEB draws from these methods, it uses a different metric to group the instances together. This metric is called the Category Utility, and was first described by Gluck and Corter [8, 9]. The COBWEB algorithm utilises a hierarchical tree to group the observed instances into concepts where traits are shared across the resident instances.

Gluck and Corter were able to show, using the category utility, similar basic level categorisation to that found within human psychological testing. Basic level categorisation, as used by Gluck and Corter, was described by Mervis and Rosch [10]. A basic level category is defined as one which is preferred to a more generalised or specific category. For example when describing an aquarium using common speech it would be preferred to describe the fish based on their colour than by a scientific or species names. Fisher showed that by using the category utility in the data mining context a probabilistic conceptual clustering algorithm could be produced, that is highly effective.

The category utility calculation takes account of each attribute in an instance, comparing it to the attributes of the instances within a given cluster, returning the utility as a measure of how much information they have in common. This attribute-value pair comparison used within COBWEB is resilient enough to produce a useful measure of likeness, and is also able to adapt to handle missing attributes within instances. Our research does not modify this calculation; for an in-depth explanation of its operation and derivation refer to Gluck and Corter [8, 9] or Fisher [5].

COBWEB is an incremental conceptual clustering algorithm and as such grows the knowledge stored within its structure one instance at a time. When a new instance is added to COBWEB, its resulting location is found by searching through the existing tree and trialling a range of options at each probable location. Each alternative is trialled, and a category utility is calculated for each result. The option with the best resulting category utility is then identified as the best choice and the new instance is stored there. A full description of the COBWEB algorithm can be found at [5, 11].

COBBIT

Since the publication of COBWEB there have been multiple authors who have modified the method to further the research. COBBIT by Kilander and Jannsson [12] is a variation of COBWEB that allows it to adapt to concept drift through the usage of a time window. The time window limits the number of instances that the knowledge hierarchy maintains within itself at any given moment. As more instances are observed and incorporated into the hierarchy the earlier observed instances are removed from the hierarchy. This allows for concept drift to occur by 'forgetting' the knowledge that was attained from older instances that were present within the hierarchy before the drift reached the current concept descriptions. COBBIT, being based upon COBWEB, is, unlike STAGGER and FLORA, also being an unsupervised learner. For this reason, and its close relationship to COBWEB, we use it to compare with our own method.

4. DynamicWEB

DynamicWEB was created to allow objects to be tracked overtime as they change, while also allowing their domain of knowledge to drift. Example domains include a person's behaviour being tracked within a security system or a person's health over a time period. While the individual person is of great interest in both scenarios, the definition of "healthy" or "a security breach" may change over time as new sicknesses or malicious activities are discovered

It is common within machine learning and data mining for latitudinal datasets to be examined where many objects are each sampled once. Patterns are extracted, models are formed, and the knowledge is gained. But there are few methods for longitudinal studies where the same person, or object, is sampled many times requiring models to be updated with the new information relating to previously observed objects. It is this problem space which DynamicWEB is investigating. By tracking each individual item as it changes, and updating the model as required, both concept and object drift are facilitated.

As a given object changes with time the instance within the DynamicWEB tree is located, updated and re-clustered with respect to its neighbours reflecting the new information gained from the most recent change. To fully understand how this was undertaken an explanation of the modifications to COBWEB will now be discussed.

The Method

COBWEB in its original form was not intended to have updates occur to the instances that were present within the tree structure. The major changes that have been made to transform COBWEB into DynamicWEB fall into two categories: the update mechanism, and the context retention process. The COBWEB concept hierarchy is sorted based on the similarity of the objects resident within it. To locate a given instance each instance would have to be examined in turn, resulting in a O(n) search time. Further, there was no provision for removing, or modifying, an instance within the tree once it was placed there. A modification of an instance within the tree itself would change the category utility of the node containing it, and any parent nodes and thus destroy the integrity of the tree. Therefore deletion and modification operations were required as well as a faster way of searching the tree. An index for the tree was implemented using an AVL tree. To maintain the integrity of the knowledge hierarchy, the update mechanism removed the target instance from the tree, updating the parent nodes recursively back to the root to subtract the knowledge from the tree, before then updating the instance with the newly observed information and reintegrating into the hierarchy. For a more in-depth discussion on these modifications see [13].

The second main addition to DynamicWEB was the use of derived attributes within a profile. These combined the knowledge from multiple observations. These derived attributes include values for mean, standard deviation, trend, and percentage deviation among others. These attributes are then selectively used within the profile to be used on a given dataset. They allowed for derived statistics to be taken from multiple observations of the target objects, preserving the historical context of the observed activity relating to a target object in comparison with the other objects that were present within the dataset. They are used both in complete form for the length of the dataset, or also in a windowed form. This latter allows for greater scalability with massive datasets, or limits the amount of data to include only the most relevant.

5. Results and Discussion

In this paper the results obtained from using DynamicWEB on two small datasets will be discussed, together with results from another small dataset, sourced from the Australian Bureau of Statistics [14]. Other results completed using DynamicWEB have been published [3] [13].

The dataset that was used to obtain the results shown in Figures 1 and 2 is the STAGGER Concepts dataset [1]. This dataset was first published along with the STAGGER method and is designed to test how fast a learner can adapt to a sudden drift that has resulted in the target class now having a totally different definition. The dataset contains two sudden drifts which result in there being three different goal class definitions to be learned over the full length of the dataset.

The performance of DynamicWEB upon this dataset is illustrated in figures 1 and 2. Within Figure 1 the mean, median and mode across 100 randomised runs completed by DynamicWEB are shown. The results are similar to, but slightly less impressive, than those published with the STAGGER and FLORA methods. However, both of these are supervised techniques, having access to the class label during the learning process, thus giving them a significant

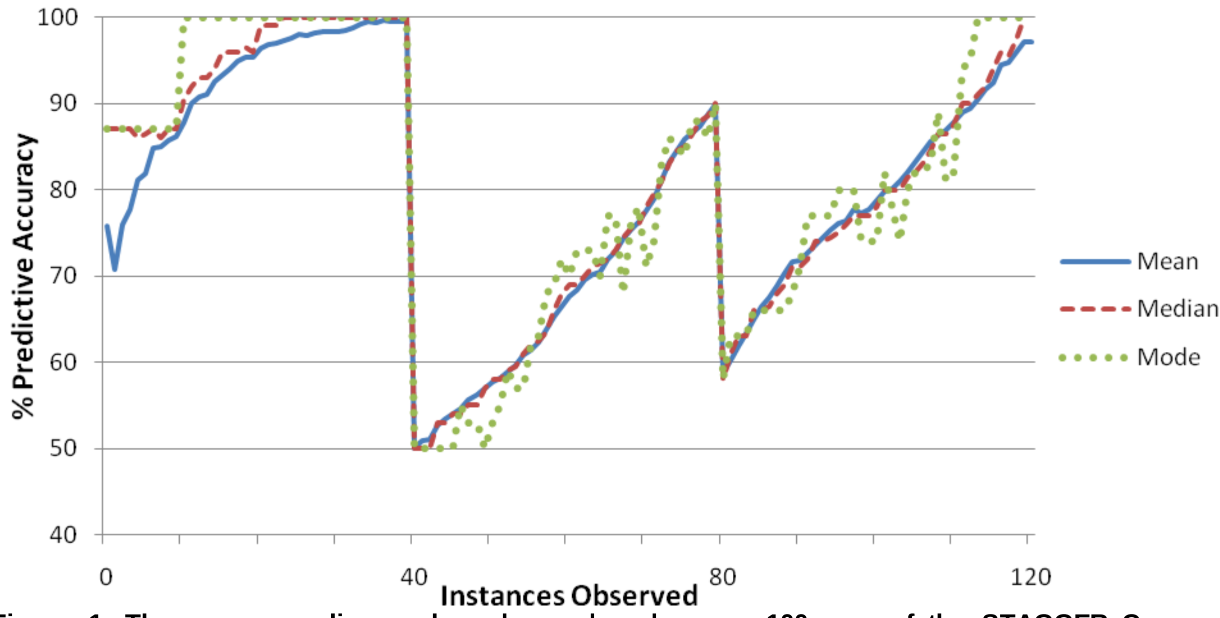


Figure 1. The mean, median and mode produced across 100 runs of the STAGGER Concepts dataset by DynamicWEB.

advantage over DynamicWEB. For all methods, learning the second concept caused the most difficulty, and in the case of FLORA meant that some instances failed to be learnt. This also occurred with DynamicWEB. The mode and the median shown within Figure 1 highlight how the second class created problems in most runs, but also how most of the runs were able to completely learn both the first and third class. The mean performance compared with the median and mode performances, upon the first concept, highlights how order-dependent the COBWEB method is, and how a poor ordering can result in significantly worse performance than is achieved by other orderings.

In Figure 2 a direct comparison is made between COBBIT and DynamicWEB using the STAGGER Concepts dataset. The comparison uses a single run of the dataset to illustrate how the two methods perform in re-learning the positive class within a domain with the same order of the observed instances. The ordering is a fairly typical ordering and is not especially good or bad when compared with

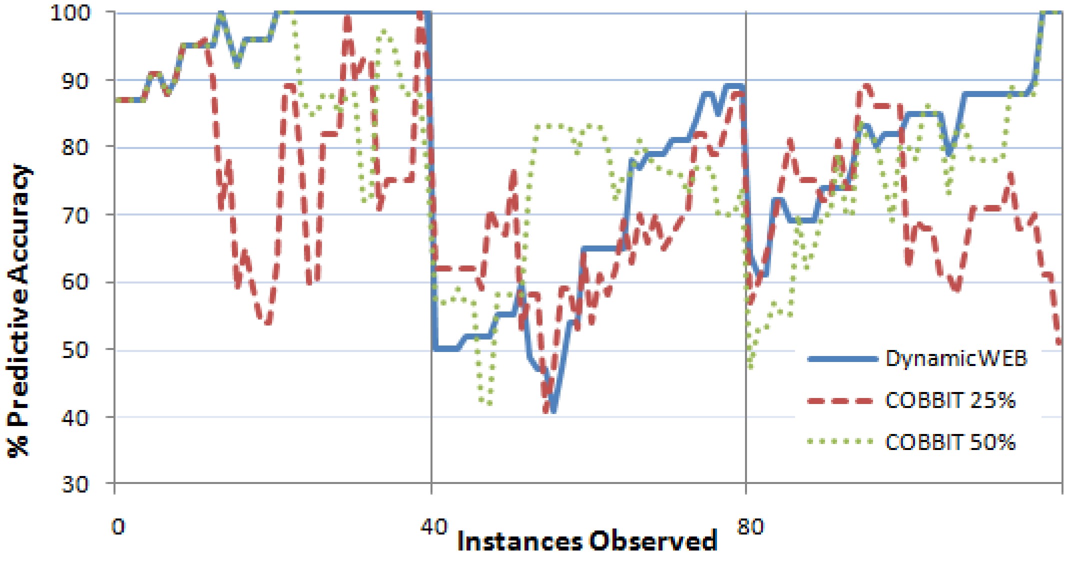
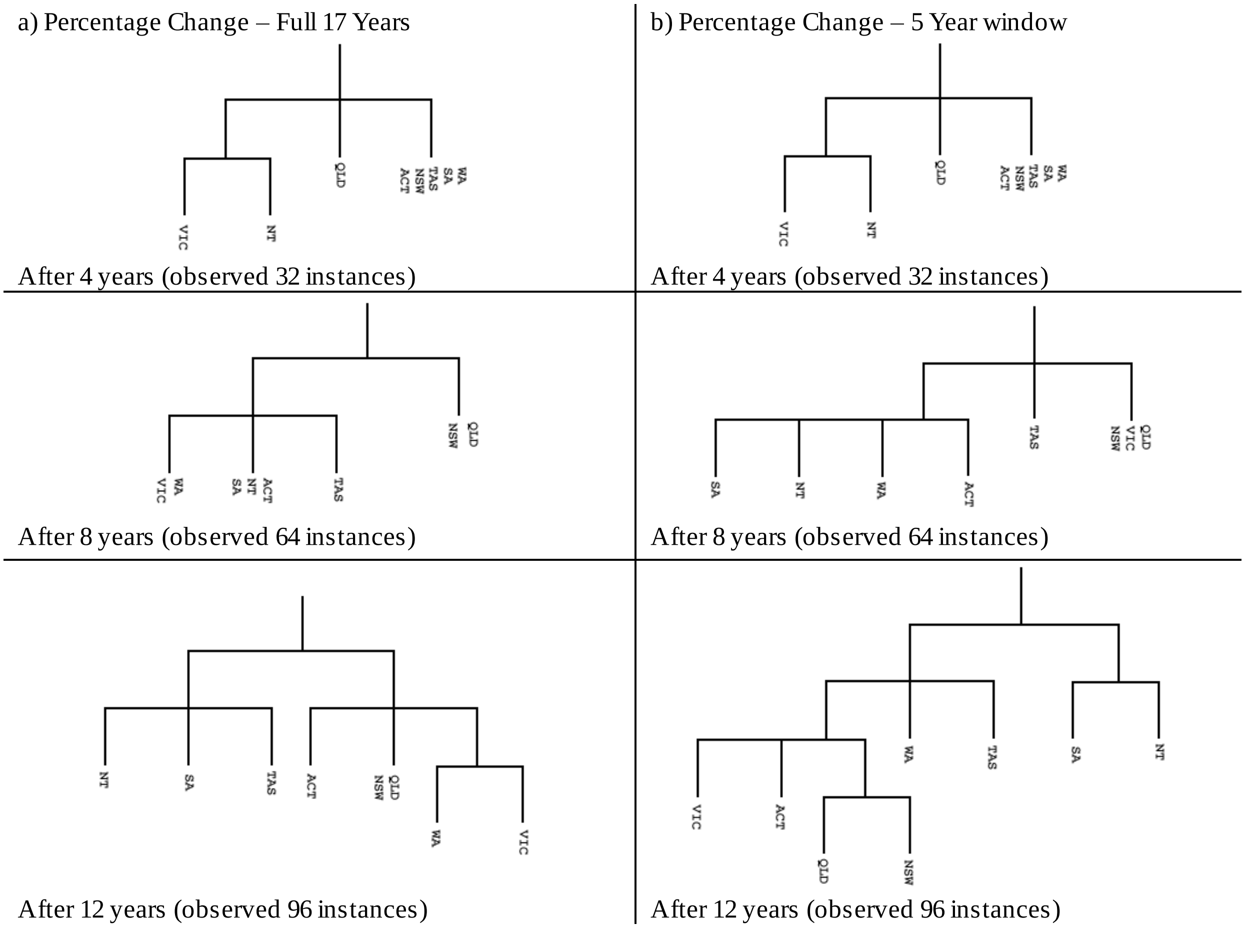


Figure 2. A direct comparison between COBBIT and DynamicWEB upon a single ordering of the STAGGER concept dataset.



a) Percentage Change – Full 17 Years

b) Percentage Change – 5 Year window

After 4 years (observed 32 instances)

After 4 years (observed 32 instances)

After 8 years (observed 64 instances)

After 8 years (observed 64 instances)

After 12 years (observed 96 instances)

After 12 years (observed 96 instances)

Figure 3. Two sets of hierarchies that were produced when examining the National Accounts dataset produced from the Australian Bureau of Statistics.

the results shown in Figure 1, and matches quite closely with the median line. DynamicWEB tends to perform better overall compared with the two COBBIT window sizes used (based upon the original authors' comments about preferred window size). However, while COBBIT does recover more quickly near the start of the new class definitions, directly after the sudden drift has taken place; it tends to suffer once the window begins to remove knowledge from the hierarchy that relates to the current target class. However, it is noted that the 50% window does perform much better than the 25% window, and is equivalent to DynamicWEB by the end of the third class definition.

Figure 3 shows two sets of knowledge hierarchies that were created using the National Accounts dataset which covers a time period of about 17 years across all the states and territories of Australia. Due to space constraints, only trees showing the knowledge hierarchy at 4-, 8- and 12-year points are shown here. The National Accounts dataset is comprised of 16 attributes that describe each state based upon several measures relating to Gross State Product and Income across several measures. These measures are in raw form, or in adjusted values that relate to the population of the state, or percentage change. The two hierarchies shown above are both based upon the same set of attributes (including three derived attributes) from this dataset. These attributes relate to the percentage change within each state across the economic measures within the dataset. However, where these structures differ is that the derived attributes used are based upon a different number of observations. The right set (b) is based upon a window of 5 observations (in this case 5 years worth of data) while the left (a) is based upon all previously observed values. Tree (a) is able to maintain the full context of the entire dataset, allowing a demographer to attempt to extract long term trends. However, by also being able to monitor the data within (b) using a restricted context, rapidly changing trends which may have been artificially smoothed by the full context of the whole dataset could also be extracted.

The addition of this ability to adjust the amount of context that is being stored and tracked by the derived attributes equips DynamicWEB with a more flexible capacity for profiling behaviour. Real world data mining domains vary widely in the amount of concept drift inherent in each domain. Being able to preserve the context within a dataset, by using derived attributes across multiple observations, is important. However, being able to control how much context is preserved is crucial to being able to extract meaningful knowledge from large datasets.

6. Conclusions and Continuing Work

Presented within this paper is a method that is not only able to adapt to concept drift when it occurs, quickly restoring decreases in predictive accuracy by updating the model, but also is also allow for individual target items to drift from one resultant class to another. This flexible learner enables the profiling of multiple objects which are observed multiple times over the course of a dataset, maintaining the context of the dataset as a whole.

In addition to the three datasets that have been discussed within this paper and the network security dataset that was also mentioned, several other datasets are being examined in this work. These datasets are of a larger scale and involve applying the machine learning technique to several data mining problems. A network performance log is being analysed along with activity logs based upon recordings made by a sensor being worn by a human whilst engaging in a range of physical activities. There is also on-going research within the network security area.

7. References

[1] Schlimmer, J.C. and R.H. Granger, *Incremental Learning from Noisy Data.* Mach. Learning, 1986. **1**(3): p. 317-354.

[2] Widmer, G. and M. Kubat, *Learning in the Presence of Concept Drift and Hidden Contexts.* Machine Learning, 1996. **23**(1): p. 69-101.

[3] Scanlan, J.D., J.S. Hartnett, and R.N. Williams, *DynamicWEB: Adapting to concept drift and object drift in COBWEB*, in *21st Australasian Joint Conference on Artificial Intelligence.* 2008: Auckland, New Zealand. p. 454-460.

[4] Michalski, R.S., *Knowledge acquisition through conceptual clustering: A theoretical framework and an algorithm for partitioning data into conjunctive concepts.* International Journal of Policy Analysis and Information Systems, 1980. **4**(3): p. 219-244.

[5] Fisher, D.H., *Knowledge Acquisition Via Incremental Conceptual Clustering* Mach. Learn. , 1987 **2** (2): p. 139-172

[6] Lebowitz, M., *Concept learning in a rich input domain:General-ization-based memory.* Machine learning:An artificial intelligence approach, 1986. **2**.

[7] Kolodner, J.L., *Reconstructive Memory: A Computer Model.* Cognitive Science, 1983. **7**: p. 281,328.

[8] Gluck, M. and J. Corter, *Information,uncertainty,and the utility of categories*, in *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*. 1985, Lawrence Erlbaum Associates: Irvine,CA. p. 283-287.

[9] Corter, J. and M. Gluck, *Explaining basic categories: Feature predictability and information.* Psychological Bulletin, 1992. **111**(2): p. 291-303.

[10] Mervis, C. and E. Rosch, *Categorization of natural objects.* Annual Review of Psychology, 1981. **32**.

[11] Gennari, J.H., P. Langley, and D. Fisher, *Models of incremental concept formation* Artif. Intell. , 1989 **40** (1-3): p. 11-61

[12] Kilander, F. and C.G. Jansson, *COBBIT - A control procdure for COBWEB in the presence of concept drift,* in *Proceedings of the European Conference on Machine Learning*, P.B. Bradzil, Editor. 1993, Springer Verlag.

[13] Scanlan, J., J. Hartnett, and R. Williams. *DynamicWEB: A method for reconnaissance activity profiling*. in *2008 International Symposium on Parallel and Distributed Processing with Applications (ISPA-08)*. 2008. Sydney Australia.

[14] Statistics, A.B.o. *Australian National Accounts: National Income, Expenditure and Product.* 2009 [cited; Available from: http://www.abs.gov.au/ausstats/abs@.nsf/mf/5206.0/.