Online Learning: Searching for the best Forgetting Strategy under Concept Drift

Ghazal Jaber[1,2,3], Antoine Cornuéjols[1,2], and Philippe Tarroux[3]

[1] AgroParisTech, UMR 518 MIA, F-75005 Paris, France
[2] INRA, UMR 518 MIA, F-75005 Paris, France
[3] Université de Paris-Sud, LIMSI, Bâtiment 508, F-91405 Orsay Cedex, France
{ghazal.jaber,antoine.cornuejols}@agroparistech.fr
philippe.tarroux@limsi.fr

Abstract. Learning from data streams in the presence of concept drifts has become an important application area. When the environment changes, it is necessary to rely on on-line learning with the capability to forget outdated information. Ensemble methods have been among the most successful approaches because they do not need hard-coded and difficult to obtain prior knowledge about the changes in the environment. However, the management of the committee of experts which ultimately controls how past data is forgotten has not been thoroughly investigated so far. This paper shows the importance of the forgetting strategy by comparing several approaches. The results lead us to propose a new ensemble method which compares favorably with the well-known CDC system based on the classical "replace the worst experts" forgetting strategy.

Key words: Online learning, ensemble methods, concept drift

1 Introduction

Recent years have witnessed the emergence of a new research area which focuses on learning from data streams in the presence of *evolving concepts*. For instance, spam filtering systems are continuously classifying incoming emails (observation $\mathbf{x}$) into spam or non-spam (label y) depending on their content. Because of changes in the spammers' strategies, corresponding to a change of the conditional distribution function $p(y|\mathbf{x})$, the filtering systems must adapt their decision rule lest they rapidly become useless.

When learning under concept drift, one central concern is to optimize a tradeoff between learning from as much data as possible, in order to get the most precise classification model, while at the same time recognizing when data points become obsolete and potentially misleading, impeding the adaptation to new trends. This is known as the *stability-plasticity dilemma.* While stability entails accumulating knowledge regarding the supposedly stationary underlying concept, plasticity, however, requires *forgetting* some or all of the old acquired knowledge in order to learn the new upcoming concept.

On-line ensemble methods have raised much interest in recent years ([1–6]). For a large part, this is due to the fact that they seem to adapt more naturally to changes in the environment than other approaches based on explicit strategies for controlling the stability-plasticity dilemma ([7]). Because of the assumed diversity of the base learners[4] in the committee, it is indeed expected that at any time some of them are ready to take over and adapt to the novelties of the environment ([8]). This diversity, however, ultimately depends on the information kept by each base learner and on the control of which learner is authorized to be part of the committee. In addition, the way the votes of the base learners are combined participates also in the overall solution to the stability-plasticity dilemma. Each of these three factors: the memory of each expert, the control of the population of experts in the committee and the weight attached to each expert in the final decision, plays a role in the way past data is taken into account by the system, what can be called the forgetting strategy.

In most ensemble methods, the first factor is implicitly governed by the second one. Each expert learns using an ever growing memory of the past data until the controller of the pool of experts decides to expel it from the committee.

In compliance with the demands of the stability-plasticity dilemma, the control strategy must be ready to introduce in the committee new base learners that will try to catch up with potential novel regularities in the environment. At the same time, it must *weaken the effect of past data that no longer represent relevant information.* There exist *two main approaches* to this problem. One is to set a threshold on the performance of the expert and to remove from the committee all experts of which the prediction performance falls below this threshold. The idea is to remove all experts that are overly biased toward obsolete regularities of the environment. This approach raises two issues. First, how to measure the prediction performance of each base learner? Second, how to set the threshold? The second family of methods does not depend on a threshold but relies instead on a perpetual renewal of the population of the committee which tends to favor a higher level of diversity. The concern here is to remove the experts that are less relevant to the current environment. Again, the question arises about the appropriate measure of performance. In addition, one must choose an insertion strategy in order to allow for the introduction of new base learners in the pool.

This paper focuses on the possible control strategies and on their impact on the performance of the system depending on the characteristics of the changing environment. We compare the two families of approaches with a special attention to the study of the deletion strategy. We do not consider the voting strategy here and keep it constant for all systems that we compare.

In the following, Section 2 describes the framework of the ensemble methods used to adapt to concept drifts while Section 3 addresses the analysis of the strategies presented above. Section 4 presents a new ensemble method using an enhanced forgetting strategy. Section 5 then reports an extensive comparison of our method with CDC ([2]) a well-known and representative ensemble method. Finally, Section 6 concludes and suggests future research directions.

[4] We use interchangeably the terms “base learner” and “expert” in this paper.

2 Ensemble methods for on-line learning

Ensemble methods for on-line learning maintain a pool of base learners $\{h_t^i\}_{1 \leq i \leq N}$, each of them adapting to the new input data, and administer this pool or ensemble thanks to a deleting strategy and an insertion one. The main components of these ensemble methods are the following:

- *Learning*: each base learner in the pool continuously adapts with new incoming data until it is removed from the pool.
- *Deletion strategy*: every τ time steps, the base learners are *evaluated* on a window of size τ_{eval}. Based on the results of the evaluation, base learners might be eliminated from the committee.
- *Insertion strategy*: every τ time steps, new base learners can be created and inserted in the pool. Each new learner starts from scratch with no knowledge of the past. It is protected from possible deletion for a duration τ_{mat}.
- *Prediction*: For each new incoming instance $\mathbf{x}_t$, the prediction $\tilde{y}_t = H(\mathbf{x}_t)$ of the committee results from a *combination* of the predictions of the individual base learners $h_t(\mathbf{x}_t)$.

Variations around this general framework lead to specific algorithms ([2–6]). The remainder of this paper will be concerned with the *deletion strategy* i.e. with the base learners selected for deletion. The insertion strategy will simply replace deleted base learners by new ones in order to keep the committee size fixed. This approach is used in most current ensemble methods ([2, 3, 5, 6]).

3 Analysis of the deletion strategies

The deletion strategy plays a key role in the adaptation process since it allows the system to forget the memory of outdated training data. In this section, we explicitly study deletion strategies based on a threshold and deletion strategies that remove the worst base learners in the committee. For the latter approach, we compare systems based on the removal of the worst expert with systems that remove experts based on other strategies.

3.1 Deletion strategies using a threshold value

A deletion strategy based on a threshold replaces base learners in the ensemble when their *prediction record*, evaluated on the window size τ_{eval}, is below a predefined threshold θ_d. When the performance is computed as the percentage of correctly classified instances on the evaluation window, only the base learners with a classification accuracy of at least $\theta_d\%$ thus remain in the committee.

This strategy leads to different behaviors depending on the characteristics of the environment. For the sake of the analysis, let us suppose that a concept drift occurs corresponding to a change in the label of *sev*% of the input space. This is called the *severity* of the concept drift ([8]). Let's suppose further that all learners in the committee have a perfect classification accuracy of 100% before the drift. We are then faced with a difficult conundrum.

If $sev << 1 - \theta_d$, the severity of the concept change is below the detection capability, and we may end up with an unchanged committee of base learners.

However, the choice of a higher threshold is loaded with two potential pitfalls. First, in case of a noisy environment, the classification accuracy of many learners may drop under the θ_d value resulting in a severely impoverished committee even though no real concept drift did happen. Second, new base learners may not be able to reach the exacting threshold before they reach the maturity age (τ_{mat}) and are therefore no longer protected from deletion.

Overall, it is difficult to set a value for a threshold without well-informed prior knowledge on the dynamics of the environment. Too low a threshold threatens the plasticity of the system, while a high one may cause havoc in the committee and prevent stability and good prediction performance. For these reasons, ensemble methods that do not rely on explicit threshold have been promoted.

3.2 Strategies that delete the worst base learner

Rather than setting a threshold for deciding which base learners to eliminate, one can encourage the diversity in the committee while preserving the best current base learners by removing the worst one every τ time steps. This should discard base learners that no longer correspond to the current state of the environment and introduce at the same time new base learners. However, a potentially vicious interaction involving the parameters τ and τ_{mat} may ruin this hope.

Let us first suppose that the period of time during which a new base learner is protected from deletion: τ_{mat} is less than τ. At each new deletion time, the newest base learner is prone to be deleted and will be if it did not have time to learn enough of the regularities in the environment. But τ cannot be too large lest the system looses any plasticity.

Suppose then that $\tau \leq \tau_{mat}$. Again the risk exists that deletion will affect only the newest learners in the committee effectively dividing the committee into a protected subset of the best and oldest base learners and a subset of the newest ones that are never able to catch up with the other ones except when the overall performance of the system has so declined that even a low prediction performance may allow a base learner to avoid elimination. And decreasing the value of τ cannot solve this problem either because if then the newly introduced base learners can survive more deletion cycles, they will still not be able to reach the performance of the established experts. Moreover, this will then tend to introduce new learners at a too high rate, threatening now to disrupt the stability and therefore the committee's performance. A question is then whether it is possible to break this poor behavior which impedes plasticity by never allowing new learners to enter the top subset.

3.3 A new deletion strategy based on a stochastic mechanism

One can soften the “eliminate the worst learner” strategy’s drawbacks by picking randomly a learner from the subset of the k ($k < N$) worst base learners. In this way, the newest base learners have a chance to learn enough of the regularities of the current environment to enter the pool of the top experts, and, at the same time, preserving the best performers. This promotes the plasticity of the system while not deteriorating its stability. The size k of the subset where base learners can be picked up to be eliminated controls the plasticity-stability trade-off.

We studied the effect of five deletion sizes (by setting the value of k) on the forgetting strategy: N, $0.75 * N$, $0.5 * N$, $0.25 * N$ and 1 (corresponding to the “always eliminate the worst base learner)”. Figure 1 shows the mean classification error depending on the different deletion sizes on the datasets of the *Line*, *SineH* and *Circle* artificial problems suggested by Minku et al. [8] to evaluate drift handling methods. The parameters were set to the values: $\tau = \tau_{mat} = \tau_{eval} = 20$, and N is 10, 20 or 30. The global prediction merely uses the prediction from the current best base learner. The base learners are decision trees (as implemented in Matlab) and all the experiments start with the same random seed so that we have the same learners at the beginning of the experiments.

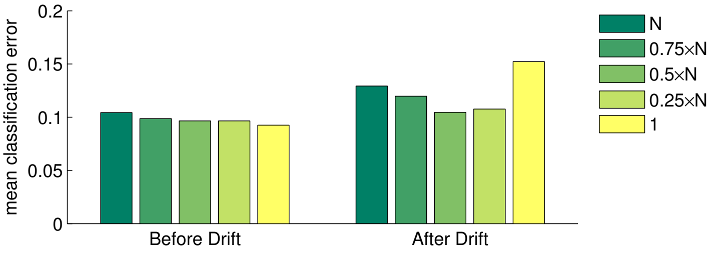

Fig. 1. The mean classification error using different deletion sizes.

Case of a stationary environment The deletion size $k = 1$ gives the best classification accuracy when learning in a stationary environment. The learners trained on the smallest windows are generally the ones that tend to be removed from the ensemble since they perform poorly compared to learners that have benefited from a large training set. Meanwhile, the remaining learners tune up their knowledge of the current concept, improving their classification record. By increasing k, the probability of removing a relatively good learner is also increased which hurts (to some extent) the classification accuracy of the ensemble.

Case of a concept drift Increasing k increases the probability of a newly added learner to survive a deletion. A large deletion size removes most of the learners from the ensemble after a concept drift. Thus, for maximum plasticity, the best deletion size is $k = N$.

The experiments suggest that k should be small enough for stability and large enough for plasticity. With a minimum deletion size ($k = 1$), the ensemble has the lowest classification error before the drift because *stability is favored over plasticity.* With a maximum deletion size however ($k = N$) the ensemble *favors plasticity over stability* which hurts the classification performance when learning stationary concepts.

A deletion size that is half the size of the ensemble ($k = 0.5 * N$) seems to correspond to a satisfactory trade-off between plasticity and stability. It yields the lowest classification error in average, before and after the concept drift.

4 DACC

We devised DACC (*dynamic adaptation to concept changes*), an online ensemble method with adaptation to possible concept drifts.

Instead of removing the worst learner of the pool, DACC selects randomly a member from the worst half of the pool and forces it to retire. In order to control the rate of deletion, we impose all the learners to be mature before a deletion operation. Therefore, $\tau = \tau_{mat}$ time steps separates two consecutive deletions.

A learner h_{bad} belonging to the worst half of the pool survives a deletion operation with a probability

$$p = \frac{N/2 - 1}{N/2} \tag{1}$$

Each time h_{bad} escapes deletion, it is given another τ_{mat} time steps of training data before the next deletion operation. The expectation of s, the number of times h_{bad} survives a deletion operation, is:

$$\begin{aligned} E[s] &= \sum_{m=1}^{\infty} mp^m(1-p) = p(1-p)\sum_{m=1}^{\infty} mp^{m-1} \\ &= p(1-p)\frac{\mathrm{d}}{\mathrm{dp}}\left(\sum_{m=1}^{\infty} p^m\right) = p(1-p)\frac{\mathrm{d}}{\mathrm{dp}}\left(\frac{p}{1-p}\right) \\ E[s] &= \frac{p}{(1-p)} \end{aligned}$$

By replacing p with its value from equation 1, we get: $E[s] = \frac{N}{2} - 1$

Increasing the life expectancy of the relatively bad learners in the pool makes DACC less exposed to cases where new learners are expelled from pool because they didn't get enough time to improve their predictive performance. In other words, the suggested forgetting strategy is less sensitive to a high deletion rate than the *replace the worst learner* strategy. A higher deletion rate entails a faster reactivity to a potential concept drift.

The new deletion strategy creates the following behavior. In periods of stability, the top base learners, being protected from deletion, will have their prediction performance improved with time as new training data are received and learnt. Their improved performance will further keep them in the top subset, allowing them to accumulate knowledge about the underlying stable target concept. The worst half of the pool will undergo periodic deletion operations. By constantly adding new learners, DACC is ready to any upcoming change, and this, without explicitly identifying a concept drift. Hence, young learners tend to be valuable during changing times while oldest learners are reliable in stable environments.

DACC follows the framework described in Section 2. The *evaluation procedure* simply counts the number of erroneous predictions on the last τ_{eval} time steps. The *deletion strategy* randomly selects one base learner from the worst half of the pool evaluated as above every $\tau = \tau_{mat}$ time steps. The *global prediction* merely uses the prediction from the current best base learner (a vote is applied in case of ties). An unmature learner does not contribute to the global prediction.

5 Experiments

We evaluated the mean classification error of DACC, CDC [2] and a learning system that does not handle drifting concepts. CDC differs from DACC in two major points. First, its *deletion strategy* evaluates the learners each time step (i.e. $\tau = 1$). A learner is removed if it is mature, if its evaluation record is below a threshold value, and if it is the worst learner in the ensemble. Secondly, the *global prediction* is the result of a weighted vote, where the weight of a learner reflects its evaluation record.

The experiments used artificial, semi-artificial and real datasets. The base learners were decision trees. The system that does not handle drifting concepts was a single decision tree trained on every training example received. CDC was evaluated with three thresholds: 0.6, 0.7 and 0.8.

The datasets used in the experiments are described in Table 1. The *artificial* datasets included Minku's et al. artificial problems [8]: *Circle*, *Line*, *SineH*, *SineV*, *Boolean* and *Plane*. Each problem consists of 9 datasets with different drift severity and speed levels (3 severities×3 speeds). The STAGGER [9] and FLORA [10] problems are among the pioneer artificial problems simulating drifting scenarios. FLORA consists of two datasets, with moderate and slow speeds of concept drifts, respectively. The *semi-artificial* datasets included IRIS and CAR [8], which are modified versions of the IRIS and CAR real datasets available in the UCI Machine Learning Repository [11]. The original real datasets were replicated several times and class labels were modified in order to simulate datastreams with multiple concept drifts. Finally, the *real dataset* was issued from the COLD database of the Saarbrücken laboratory [12], a benchmark for vision-based localization systems. It contains sequences of images recorded by a mobile robot under different variations of illumination and weather: sunny, cloudy and night. We worked on the dataset captured in sunny conditions. Images were first pre-processed into a 128-dimensional space using the Self-Organizing Map described in [13].

Table 1 reports the mean classification error of the different approaches along with the preset parameter values. For Minku's artificial problems, the error was averaged over the 9 different datasets of the corresponding problem. For STAG-

Table 1. The mean classification error of the different approaches along with the predefined parameter values, with $\tau_{mat} = \tau_{eval}$. We detail the size of each dataset, the size of its feature space, the number of classes and the dataset type: A, S, R, for artificial, semi-artificial and real, respectively.

Datasets					**mean classification error**					**Settings**
name	size	#feat.	#class.	type	DACC	CDC 0.8	CDC 0.7	CDC 0.6	NoDriftH.	τ_{eval}, N
CAR	1296	6	2	S	14.66	23.47	15.14	**13.98**	40.77	20,20
IRIS	338	4	4	S	**15.33**	15.38	18.34	20.71	34.62	20,20
Circle	2000	2	2	A	**6.92**	35.66	8.09	8.73	12.56	20,20
Line	2000	2	2	A	**3.72**	8.12	4.64	7.01	10.52	20,20
Boolean	1000	3	2	A	**2.75**	3.85	4.17	4.47	17.63	20,20
SineH	2000	2	2	A	**13.0**	47.81	20.66	13.26	21.25	20,20
SineV	2000	2	2	A	**4.32**	7.98	5.15	5.85	11.08	20,20
Plane	1000	11	2	A	20.87	20.55	**18.01**	19.95	27.23	20,20
STAGGER	120	3	2	A	**14.33**	16.08	17.75	20.25	32.52	10,10
FLORA-M	500	6	2	A	**5.34**	10.19	6.32	6.02	16	10,10
FLORA-S	500	6	2	A	**7.7**	12.46	9.21	9	18.9	10,10
COLD	753	128	4	R	**6.04**	7.83	8.23	9.16	35.46	10,30

GER and FLORA problems, the error was averaged over 10 instantiations of the datasets. For IRIS, CAR and the COLD datasets, the error was averaged over 10 runs on the same dataset.

DACC has the smallest classification error in all cases, except for the CAR and *Plane* datasets. The results on the CAR dataset suggest that a deletion threshold of 0.6 is adapted to the CAR learning problem. Hence, removing learners with a classification accuracy smaller than 60% allows the ensemble to adapt to the simulated concept changes. For the *Plane* dataset, the difference between DACC and CDC 0.7 is likely due to the noise in the *Plane* dataset. Generally, the use of a max function for the global prediction (as in DACC) instead of a weighted combination (as in CDC) affects the predictive accuracy in noisy environments. It results in this case in a higher classification error for DACC by a margin of 2.86%.

6 Conclusion and Future Work

This paper presents an analysis of two main forgetting strategies used by existing online ensemble methods to adapt to concept drifts: (a) deleting experts with poor predictive performance, according to a preset threshold value, and (b) deleting periodically the worst expert in the ensemble. The ensuing analysis lead to the definition of a new approach (DACC) to handle concept drifts.

The analysis shows that the forgetting strategy r(a) equires prior knowledge on the dynamics of the environment in order to choose an adapted threshold value, while strategy (b) may result in unwanted behavior, affecting the ability of the ensemble to adapt to new trends.

DACC deletes periodically one expert chosen randomly from the worst half of the ensemble. According to our study, this strategy corrects unexpected behaviors of the latter forgetting strategy. Empirical comparisons with CDC, a representative method based on the former forgetting strategy, show that DACC overcomes the difficulty of finding the appropriate threshold, and this on a large variety of concept drifts, with several levels of severity and speed.

For future work, we plan to study another key component of the forgetting strategy: the way experts are weighted and the way their decisions are combined in the ensemble's final decision.

References

1. Wang, H., Fan, W., Yu, P. S., Han, J.: Mining concept-drifting data streams using ensemble classifiers. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 226–235. ACM (2003)
2. Stanley, K. O.: Learning concept drift with a committee of decision trees. In: Informe técnico: UT-AI-TR-03-302, Department of Computer Sciences, University of Texas at Austin, USA (2003)
3. Tsymbal, A., Pechenizkiy, M., Cunningham, P., Puuronen, S.: Dynamic integration of classifiers for handling concept drift. In: Information Fusion, 9(1), pp. 56–68 (2008)
4. Scholz, M., Klinkenberg, R.: An ensemble classifier for drifting concepts. In: Proceedings of the Second International Workshop on Knowledge Discovery in Data Streams. Porto, Portugal (2005)
5. Bifet, A., Holmes, G., Pfahringer, B., Kirkby, R., Gavaldà, R.: New ensemble methods for evolving data streams. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM (2009)
6. Kolter, J. Z., Maloof, M.A.: Dynamic weighted majority: A new ensemble method for tracking concept drift. In: Data Mining, 2003. ICDM 2003. Third IEEE International Conference, pp. 123-130 (2013)
7. Karnick, M., Ahiskali, M., Muhlbaier, M. D., Polikar, R.: Learning concept drift in nonstationary environments using an ensemble of classifiers based approach. In: Neural Networks IJCNN 2008. IEEE International Joint Conference, pp. 3455–3462. IEEE (2008)
8. Minku, L. L., White, A. P., Yao, X.: The impact of diversity on online ensemble learning in the presence of concept drift. In: Knowledge and Data Engineering, IEEE Transactions on 22.5, pp. 730–742 (2010)
9. Schlimmer, J. C., Granger, R.: Beyond incremental processing: Tracking concept drift. In: Proceedings of the Fifth National Conference on Artificial Intelligence, vol. 1 (1986)
10. Widmer, G., Kubat, M.: Learning in the presence of concept drift and hidden contexts. In: Machine learning 23, no. 1, pp. 69–101 (1996)
11. UCI Machine Learning Repository, `http://archive.ics.uci.edu/ml/`
12. The COLD Database, `http://www.cas.kth.se/COLD/`
13. Guillaume, H., Dubois, M., Emmanuelle, F., Tarroux, P.: Temporal Bag-of-Words-A Generative Model for Visual Place Recognition using Temporal Integration. In: VISAPP-International Conference on Computer Vision Theory and Applications-2011 (2011)