

Performance Metrics for Model Fusion in Twitter Data Drifts

Joana Costa^{1,2(✉)}, Catarina Silva^{1,2}, Mário Antunes^{1,3},
and Bernardete Ribeiro²

¹ School of Technology and Management,
Polytechnic Institute of Leiria, Leiria, Portugal
{joana.costa,catarina,mario.antunes}@ipleiria.pt

² Department of Informatics Engineering,
Center for Informatics and Systems of the University of Coimbra (CISUC),
Coimbra, Portugal

{joanamc,catarina,bribeiro}@dei.uc.pt

³ Center for Research in Advanced Computing Systems, INESC-TEC,
University of Porto, Porto, Portugal
mantunes@dcc.fc.up.pt

Abstract. Ensemble approaches have revealed remarkable abilities to tackle different learning challenges, namely in dynamic scenarios with concept drift, e.g. in social networks, as Twitter. Several efforts have been engaged in defining strategies to combine the models that constitute an ensemble. In this work, we investigate the effect of using different metrics for combining ensembles' models, specifically performance-based metrics. We propose five performance combining metrics, having in mind that we may take advantage of diversity in classifiers, as their individual performance takes a leading role in defining their contribution to the ensemble. Experimental results on a Twitter dataset, artificially timestamped, suggest that using performance metrics to combine the models that constitute an ensemble can introduce relevant improvements in the overall ensemble performance.

Keywords: Ensembles · Twitter · Dynamic environments

1 Introduction

Nowadays most learning problems demand dynamic models, which can adapt to new circumstances as they emerge. Paradigmatic to this setting are social networks scenarios, as Twitter, where new information appears all the time. Different approaches have been pursued with such goals, like ensemble systems for classification problems, presented and discussed in this work.

Ensembles of classifiers integrate multiple classifiers to classify each example with the aim of improving classification performance. There are many approaches for ensemble of classifiers, such as boosting [1], bagging [2], or random forests [3],

but their original form is usually applied in static environments. However, ensembles are specially adequate to tackle dynamic evolving settings, given their modular nature, and different studies and approaches have been pursued [4, 5].

In this work, we investigate the effect of using different metrics for combining ensembles' classifiers, specifically performance-based metrics. We propose a framework where the diversity in classifiers is explored using their individual performance as driver for the definition of their weight in the ensemble. The approach is then embedded with the importance that weight asymmetry performance metrics has in boosting the model fusion overall success. Five performance evaluation metrics are then proposed.

The rest of the paper is organized as follows. In Sect. 2 we introduce background concepts and state of the art on model fusion and social networks, focusing on Twitter approaches. In Sect. 3 we introduce our approach for model fusion using different metrics to evaluate their individual performance in order to define their contribution to the ensemble. In Sect. 4 we present the experimental setup with the construction of the benchmark dataset and the evaluation metrics. In Sect. 5 we present and analyse the results obtained by comparing the metrics for combining models in an ensemble and, finally, conclusions and future work.

2 Background

2.1 Ensembles

Ensembles are cutting-edge solutions to many different learning challenges. Different researchers have been studying ensembles and their applications in various fields [4, 6–8].

Classifier committees or ensembles are based on the idea that, given a task that requires expert knowledge, k experts (baseline classifiers) may perform better than one, if their individual judgements are appropriately combined. A classifier committee is then characterized by (i) a choice of k classifiers, and (ii) a choice of a combination function, sometimes denominated a *voting algorithm*. The classifiers should be as independent as possible to guarantee a large number of inductions on the data. By using different classifiers to exploit diverse patterns of errors to make the ensemble better than just the sum (or average) of the parts, we may obtain a gain from synergies between the ensemble classifiers.

Ensembles are used in different setting like novelty detection [9], and though the simplest combination function is just a majority voting mechanism with an odd number of baseline classifiers, different fusion mechanisms have been proposed, namely: average, minimum, maximum, median, majority vote, and oracle [10].

In [11, 12] two approaches of incremental learning of concept drift in non-stationary environments are presented. The authors describe ensemble-based approaches of classifiers for incrementally learning from new data drawn from a distribution that changes in time and generates a new classifier using each additional dataset that becomes available from the changing environment.

2.2 Social Networks: Twitter Case Study

Social networks are paradigmatic examples of dynamic environments. Specifically, Twitter is such a case where drift phenomena commonly occur in a text-base scenario. Twitter is a micro-blogging service where users post text messages up to 140 characters, also known as tweets. Twitter is also responsible for the popularization of the concept of hashtag, a single word started by the symbol “#” that is used to classify the message content and to improve search capabilities. hashtags can also be used as a classification label. If we can classify a tweet based on a set of hashtags, we are able to suggest an hashtag for a new given tweet, bringing a wider audience into discussion [13], spreading an idea [14], get affiliated with a community [15], or bringing together other Internet resources [16].

This case study aims to classify Twitter messages. A Twitter classification problem can be described as a multi-class problem that can be cast as a time series of tweets. It consists of a continuous sequence of instances, in this case, Twitter messages, represented as $\mathcal{X} = \{x_1, \dots, x_t\}$, where x_1 is the first occurring instance and x_t the latest. Each instance occurs at a time, not necessarily in equally spaced time intervals, and is characterized by a set of features, usually words, $\mathcal{W} = \{w_1, w_2, \dots, w_{|\mathcal{W}|}\}$. Consequently, instance x_i is denoted as the feature vector $\{w_{i1}, w_{i2}, \dots, w_{i|\mathcal{W}|}\}$.

We have used a classification strategy previously introduced in [17]. Assuming x_i is a labelled instance it is represented as the pair (x_i, y_i) , being $y_i \in \mathcal{Y} = \{y_1, y_2, \dots, y_{|\mathcal{Y}|}\}$ the class label for instance x_i , or the hashtag that labels the Twitter message x_i .

3 Proposed Approach

Figure 1 depicts the ensemble model that underpins the proposed framework of metrics for combining ensembles. The model can be divided in three parts, from top to bottom: (i) models’ construction; (ii) learning process; (iii) models’ combination.

The construction of the models is carried out by defining time-windows and learning models for each time-window. Different scenarios can be constructed, i.e., the exact examples that are considered in each time-window depend on the specific approach. The simplest approach is to consider just the timestamp of the example, but more elaborate approaches may consider the relevance of the example or the effort for it to be learned [18].

The learning process focuses on the definition of the k baseline classifiers. Notice that in dynamic environments, the ensemble must adapt to deal with changes usually dependent of hidden contexts. One of the major challenges in dynamic environments is the amount of data, specially when dealing with streams. It is sometimes infeasible to store all the previously seen data, but it may carry substantial information for future use. Hence, not all previously constructed models are kept in the ensemble and, in the learning process the

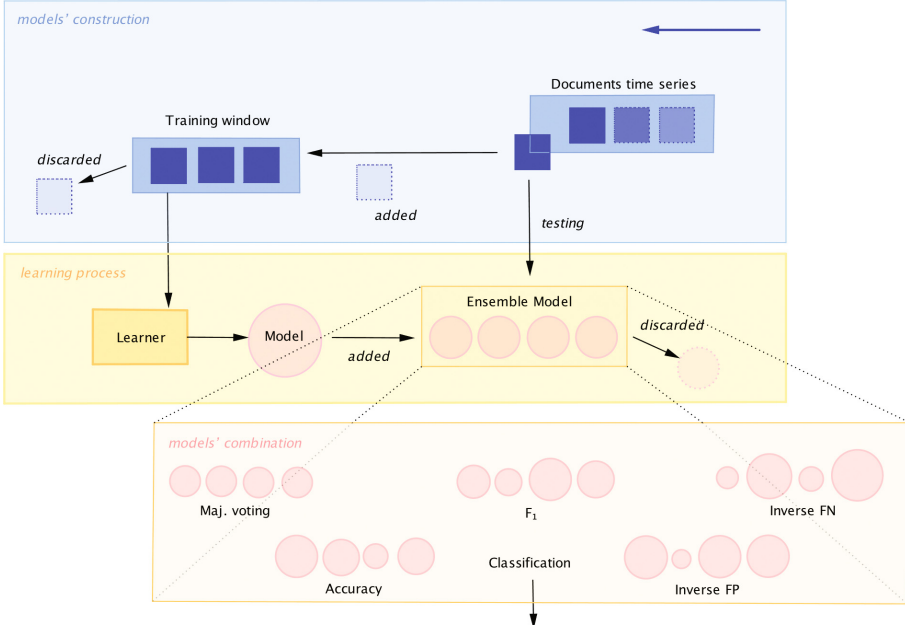


Fig. 1. Proposed ensemble model ($k = 4$) with combining metrics

decision of which ones should be kept (or added) and which ones should be discarded takes place [19,20].

The framework proposed in the paper uses a combination of models with different performance metrics. The underpinning idea behind the deployed framework is to test and evaluate different strategies to combine baseline models into an ensemble. By doing this we aim to increase the classification performance, as we may tackle the problem of not being able to store all the previously unseen examples.

To evaluate a binary classification task, TP, FP, TN and FN values are obtained and then a set of performance metrics can be defined: error rate ($\frac{FN+FP}{TP+TN+FP+FN}$), accuracy ($\frac{TP+TN}{TP+TN+FP+FN}$), recall ($R = \frac{TP}{TP+FN}$), and precision ($P = \frac{TP}{TP+FP}$), as well as combined measures, such as, $F_1 = \frac{2 \times P \times R}{P+R}$ [21]. As can be gleaned from Fig. 1, the proposed metrics are: majority voting, accuracy, inverse FP ($\frac{1}{FP}$), inverse FN ($\frac{1}{FN}$), and F_1 .

Considering the proposed approach and the fact that we are working with a time series in a “one-against-all” strategy, we will have a classifier for each batch of the time series that is composed by $|Y|$ binary classifiers, being $|Y|$ the collection of possible labels. To perceive the performance of the classification for each drift pattern, we will consider all the binary classifiers that were created in all the time series batches. To evaluate the performance obtained across time, we will average the obtained results. Two conventional methods are widely used, specially in multi-label scenarios, namely macro-averaging and micro-averaging.

Macro-averaged performance scores are obtained by computing the scores for each learning model in each batch of the time series and then averaging these scores to obtain the global means. Differently, micro-averaged performance scores are computed by summing all the previously introduces contingency matrix values (a, b, c and d), and then use the sum of these values to compute a single micro-averaged performance score that represents the global score.

The metrics we are proposing are based on the performance of each model whenever a new example arrives. As a consequence, if a model is unable to correctly classify examples in a given moment, its performance metrics will decrease, or even be null, excluding the contribution of the model to the ensemble in the subsequent moments. However, if in another moment, the model regains the ability to correctly classify the example, the increase of its performance will allow it to contribute again. This is particularly important in dynamic environments where concepts can appear and reappear.

4 Experimental Setup

The Twitter dataset we have defined to evaluate and validate our strategy was carried out by defining 10 different hashtags that represent different drifts, based on the assumption that they would denote mutually exclusive concepts.

The Twitter API (<https://dev.twitter.com/>) was then used to request public tweets containing the defined hashtags. The requests were submitted between 28 December 2014 and 21 January 2015 and tweets were only considered if the user language was defined as English. We have requested more than 75.000 tweets with the given hashtags. The hashtag was then removed from the tweet and was exclusively used as the document label. The tweets were then labelled for classification purposes, and were used by their appearing order in the public feed. Our final dataset comprises 34.240 tweets.

Table 1. Mapping between type of drift and hashtag.

Drift	Hashtag
Sudden #1	#syrisa
Gradual #1	#isis
Incremental #1	#android
Reoccurring	#realmadrid
Normal #2	#sex
Sudden #2	#airasia
Gradual #2	#bieber
Incremental #2	#ferrari
Normal #1	#jobs
Normal #3	#nfl

Table 1 presents the hashtags and the corresponding type of drift represented by each one. This correspondence was arbitrary and does not correspond to any real occurrence in a real Twitter scenario, since as stated above, no information is known about the occurrence of drifts in Twitter. The final dataset was constructed using DOTS (Drift Oriented Tool System), a free drift oriented framework we have developed to dynamically create datasets with drift [22]. It can be freely downloaded at <http://dotspt.sourceforge.net/>. The evaluation of our approach was done by the previously described dataset and using Support Vector Machines (SVM) [23].

5 Experimental Results

In this section we evaluate the performance obtained with the Twitter data set using the approach described in Sect. 3. Besides a majority voting strategy, five performance metrics were used to combine the ensemble model: accuracy, F_1 measure, the inverse of false positives and the inverse of false negatives. The majority voting strategy is used as baseline, as all models contribute equally to the final decision of the ensemble, despite their previous performance.

Table 2 summarises the performance results obtained by classifying the dataset, considering the micro-averaged F_1 measure.

Table 2. Micro-averaged F_1 for different combining metrics

Drift	Performance metrics for model fusion				
	Maj. voting	Accuracy	F_1	Inverse FP	Inverse FN
Sudden #1	74.80%	79.67%	87.95%	89.12%	89.15%
Sudden #2	87.80%	89.12%	92.76%	93.20%	93.17%
Gradual #1	52.55%	54.82%	65.72%	68.27%	68.27%
Gradual #2	62.21%	65.20%	76.83%	78.93%	78.93%
Incremental #1	88.58%	88.89%	89.50%	91.68%	91.80%
Incremental #2	77.21%	77.31%	78.08%	80.31%	80.13%
Reoccurring	35.33%	36.75%	59.76%	63.95%	64.53%
Normal #1	70.89%	70.95%	71.26%	73.01%	73.01%
Normal #2	90.49%	90.51%	90.55%	90.81%	90.90%
Normal #3	81.52%	81.63%	81.97%	82.10%	82.08%
Micro-averaged F_1	78.27%	79.39%	82.99%	84.36%	84.39%

Analysing the table we can observe that using different metrics to combine the ensemble can lead to different performance results, considering the Twitter classification problem. Globally we can achieve a 6% increase in the F_1 measure, when comparing the use of a majority voting strategy with a performance based strategy like inverse FN.

It is particularly important to note that this performance increase is observed in all different types of drifts, despite their nature. We have also observed that the obtained results were achieved because the number of false negatives was reduced when using metrics based on the performance. Though this might be problem dependent, it is also relevant to pinpoint.

Table 3 summarises the performance results obtained by classifying the dataset, considering the micro-averaged Recall measure. Recall is highly dependent on the true positives, and the increased show us that using different metrics can reduce false negatives and consequently increase the true positives. The reduction of false negatives is, consequently, responsible for the increase of the F_1 results presented in Table 2, as precision is not significantly affected when using different combining strategies.

Table 3. Micro-averaged Recall for different combining metrics

Drift	Performance metrics for model fusion				
	Maj. voting	Accuracy	F_1	Inverse FP	Inverse FN
Sudden #1	59.78%	66.25%	78.53%	80.48%	80.52%
Sudden #2	78.36%	80.50%	86.61%	87.42%	87.39%
Gradual #1	35.67%	37.79%	49.00%	51.92%	51.92%
Gradual #2	45.54%	48.79%	62.67%	65.54%	65.54%
Incremental #1	79.87%	80.40%	81.40%	85.01%	85.24%
Incremental #2	63.67%	63.79%	64.83%	67.92%	67.68%
Reoccurring	21.47%	22.53%	42.67%	47.13%	47.73%
Normal #1	54.95%	55.03%	55.39%	57.53%	57.54%
Normal #2	82.93%	82.97%	83.03%	83.56%	83.68%
Normal #3	68.91%	69.07%	69.56%	69.80%	69.77%
Micro-averaged recall	64.55%	66.08%	71.18%	73.25%	73.29%

6 Conclusions

In this paper we evaluate the use of performance metrics to combine models that constitute an ensemble in a Twitter classification problem. The main idea is to boost the classification performance of the ensemble model by combining its models based on their previous performance, and thus giving more weight to the contribution of a best performer model when compared to a less performer one.

We have used a Twitter case study to evaluate our approach. Since it is not known which types of drift occur in the context of social networks, and particularly in Twitter, we have also simulated different types of drift in a dataset generated artificially with real tweets to evaluate and validate our strategy.

The results revealed the usefulness of our strategy, as using different performance-based metrics led to the improve by 6%. This result was obtained

considering the micro-averaged F_1 , when comparing to the baseline approach, a majority voting strategy, where all models contribute equally to the final decision of the ensemble, despite their previous performance.

We may also conclude that results obtained and the improvement observed are independent from the drift pattern the class represents, and thus can be applied in different dynamic scenarios. A more suited metric can better weight the best performer models and thus increase the ensemble overall performance, as less performer models can cease their contribution. Future work will include more complex performance metrics. More efforts are needed to understand if longevity can also be included in the contribution of a model in the ensemble and if a pruning strategy is worth applying.

Acknowledgment. It is also financed by national funding via the Foundation for Science and Technology and by the European Regional Development Fund (FEDER), through the COMPETE 2020 - Operational Program for Competitiveness and Internationalization (POCI).

References

1. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **55**(1), 119–139 (1997)
2. Breiman, L.: Bagging predictors. *Mach. Learn.* **24**(2), 123–140 (1996)
3. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
4. Bagul, R.D., Phulpagar, B.D.: Survey on approaches, problems and applications of ensemble of classifiers. *Int. J. Emerg. Trends Technol. Comput. Sci.* **5**(1), 28–30 (2016)
5. Ditzler, G., Polikar, R.: Incremental learning of concept drift from streaming imbalanced data. *IEEE Trans. Knowl. Data Eng.* **25**(10), 2283–2301 (2013)
6. Tabassum, N., Ahmed, T.: A theoretical study on classifier ensemble methods and its applications. In: 3rd International Conference on Computing for Sustainable Global Development, pp. 67–78 (2016)
7. Ren, Y., Zhang, L., Suganthan, P.N.: Ensemble classification and regression - recent developments, applications and future directions. *IEEE Comput. Intell. Mag.* **1**(1), 41–43 (2016)
8. Ponti Jr., M.P.: Combining classifiers: from the creation of ensembles to the decision fusion. In: 24th Conference on Graphics, Patterns and Images, pp. 1–10 (2011)
9. Faria, E., de Carvalho, A., Gonçalves, I., Gama, J.: Novelty detection in data streams. *Artif. Intell. Rev.* **45**(2), 235–269 (2016)
10. Kuncheva, L.: A theoretical study on six classifier fusion strategies. *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(2), 281–286 (2002)
11. Elwell, R., Polikar, R.: Incremental learning of concept drift in nonstationary environments. *IEEE Trans. Neural Netw.* **22**, 1517–1531 (2011)
12. Karnick, M., Muhlbaier, M.D., Polikar, R.: Incremental learning in non-stationary environments with concept drift using a multiple classifier based approach. In: International Conference on Pattern Recognition, pp. 1–4 (2008)
13. Johnson, S.: How Twitter will change the way we live. *Time Mag.* **173**, 23–32 (2009)

14. Tsur, O., Rappoport, A.: What's in a hashtag?: content based prediction of the spread of ideas in microblogging communities. In: Proceedings of the 5th International Conference on Web Search and Data Mining, pp. 643–652 (2012)
15. Yang, L., Sun, T., Zhang, M., Mei, Q.: We know what @you #tag: does the dual role affect hashtag adoption? In: Proceedings of the 21st International Conference on World Wide Web, pp. 261–270 (2012)
16. Chang, H.-C.: A new perspective on Twitter hashtag use: diffusion of innovation theory. In: Proceedings of the 73rd Annual Meeting on Navigating Streams in an Information Ecosystem, pp. 85:1–85:4 (2010)
17. Costa, J., Silva, C., Antunes, M., Ribeiro, B.: Defining semantic meta-hashtags for Twitter classification. In: Tomassini, M., Antonioni, A., Daolio, F., Buesser, P. (eds.) ICANNGA 2013. LNCS, vol. 7824, pp. 226–235. Springer, Heidelberg (2013). doi:[10.1007/978-3-642-37213-1_24](https://doi.org/10.1007/978-3-642-37213-1_24)
18. Costa, J., Silva, C., Antunes, M., Ribeiro, B.: Choice of best samples for building ensembles in dynamic environments. In: Jayne, C., Iliadis, L. (eds.) EANN 2016. CCIS, vol. 629, pp. 35–47. Springer, Cham (2016). doi:[10.1007/978-3-319-44188-7_3](https://doi.org/10.1007/978-3-319-44188-7_3)
19. Costa, J., Silva, C., Antunes, M., Ribeiro, B.: The impact of longstanding messages in micro-blogging classification. In: International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2015)
20. Costa, J., Silva, C., Antunes, M., Ribeiro, B.: Concept drift awareness in Twitter streams. In: Proceedings of the 13th International Conference on Machine Learning and Applications, pp. 294–299 (2014)
21. Sokolova, M., Lapalme, G.: A systematic analysis of performance measures for classification tasks. *Inf. Process. Manage.* **45**(4), 427–437 (2009)
22. Costa, J., Silva, C., Antunes, M., Ribeiro, B.: DOTS: drift oriented tool system. In: Arik, S., Huang, T., Lai, W.K., Liu, Q. (eds.) ICONIP 2015. LNCS, vol. 9492, pp. 615–623. Springer, Cham (2015). doi:[10.1007/978-3-319-26561-2_72](https://doi.org/10.1007/978-3-319-26561-2_72)
23. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1999)

<http://www.springer.com/978-3-319-58837-7>

Pattern Recognition and Image Analysis

8th Iberian Conference, IbPRIA 2017, Faro, Portugal,

June 20-23, 2017, Proceedings

Alexandre, L.A.; Salvador Sánchez, J.; Rodrigues, J.M.F.

(Eds.)

2017, XVI, 549 p. 180 illus., Softcover

ISBN: 978-3-319-58837-7