

## Twitter classification: are some examples better than others?

Joana Costa<sup>12</sup>

joana.costa@ipleiria.pt, joanamc@dei.uc.pt

Catarina Silva<sup>12</sup>

catarina@ipleiria.pt, catarina@dei.uc.pt

Mário Antunes<sup>13</sup>

mario.antunes@ipleiria.pt, mantunes@dcc.fc.up.pt

Bernardete Ribeiro<sup>2</sup>

bribeiro@dei.uc.pt

<sup>1</sup> School of Technology and Management  
Polytechnic Institute of Leiria, Portugal

<sup>2</sup> CISUC - Department of Informatics Engineering  
University of Coimbra, Portugal

<sup>3</sup> Center for Research in Advanced Computing Systems  
INESC-TEC, University of Porto, Portugal

### Abstract

One of the major challenges in dynamic environments is the amount of data, specially when dealing with streams. It is sometimes unfeasable to store all the previously seen data, despite the fact that it may carry substantial information for future use. Two questions arise: (i) How is it possible to enhance the input examples? (ii) Are there examples better than others, that thus should be kept for future use?

In this paper we propose a method that determines the most relevant examples by analysing their behaviour when defining separating planes between classes. We have tested our approach in a Twitter scenario and results show that keeping those examples improves the classification performance.

### 1 Introduction

Social networks have settled definitely in the daily routine of Internet users. They have also gained increasing importance and are being widely studied in many fields of research over the last years, such as computer, social, political, business and economical sciences. With millions of daily users, they are an important source of information and learning in those environments can have multiple benefits, like market sensing, recommendation, event detection, sentiment analysis, among others.

Considering their potential in information spread, it is imperative to find learning strategies able to learn in social networks. However, their dynamic nature, requires specific learning approaches. Differently from the commonly used approaches, effective learning in such scenarios requires a learning algorithm with the ability to detect context changes without being explicitly informed about them, quickly recovering from those context changes and adjusting hypothesis to new contexts. Multiple drift patterns were identified by Zliobaite [1], namely sudden, gradual, incremental, and reoccurring.

The focus of our work is on the Twitter social media platform ([www.twitter.com](http://www.twitter.com)), more precisely on applying learning and classification strategies to learn in the presence of different types of variations of context (*drift*) through time [2,3]. We have used an artificial dataset with Twitter messages that simulates those drift patterns.

### 2 Background

Twitter stream constitutes a paradigmatic example of a text-based scenario where drift phenomena occur commonly. *Twitter* is a micro-blogging service where users post text-based messages up to 140 characters, also known as *tweets*.

*Twitter* is also responsible for the popularization of the concept of *hashtag*. An *hashtag* is a single word started by the symbol “#” that is used to classify the message content and to improve search capabilities. Besides improving search capabilities, *hashtags* have been identified as having multiple and relevant potentialities, like those described in [4].

Considering the importance of the *hashtag* in Twitter, it is relevant to study the possibility of evaluating message contents in order to predict its *hashtag*. If we can classify a message based on a set of *hashtags*, we are able to suggest an *hashtag* for a given *tweet*. Social networks can be seen as a dynamic and non-stationary environment, in which information is produced by users in a timely order. Time plays a crucial role in Twitter information processing, as past events can give important insights to understand how previously seen information is relevant to improve learning and classification of future unseen and related events. In that sense, learning strategies would be able to learn in dynamic environments and apply innovative strategies to deal with a “recent memory” of past events, in order to better identify future and previously unseen ones. There can be several approaches to tackle dynamic environments [5]: instance selection, instance weighting and ensemble learning. A review of concept drift applied to intrusion detection is presented in [6].

### 3 Proposed Approach

#### 3.1 Twitter classification problem

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  $\{w_{i1}, w_{i2}, \dots, w_{i|\mathcal{W}|}\}$ . When  $x_i$  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  $x_i$ .

We have used a classification strategy previously introduced in [7], where the Twitter message *hashtag* is used to label the content of the message, which means that  $y_i$  represents the *hashtag* that labels the Twitter message  $x_i$ .

Notwithstanding being a multi-class problem in its essence, it can be decomposed in multiple binary tasks in a one-against-all binary classification strategy. In this case, a classifier  $h'$  is composed by  $|\mathcal{Y}|$  binary classifiers.

#### 3.2 Learning Models

In [3] we have studied the impact of longstanding examples in future classification time-windows. The rationale of the presented idea was to store previously seen examples for a period of time regardless the effect they might have as a solo example. Differently from that approach, we are now proposing to choose examples based on the effect they might have individually.

Our baseline model, created for comparison purposes, proposes to store all the information gathered by storing models and combining them as an ensemble. For each time-window, a classifier is trained and stored. When a new collection of documents, in the subsequent time-window, occurs, all the previously trained classifiers are loaded, and the system will classify the newly seen examples. The prediction function of the ensemble, composed by the set of classifiers already created, is a combined function of the outputs of all the considered classifiers. A majority voting strategy where each model participates equally is then put forward. The documents of the previously seen time-windows are not stored in this approach even though the possible learning information is stored along in the classifier trained immediately after it.

We then propose an ensemble learning model, the reinforced model. The main difference is that we define a collection of documents that contains all the classification errors that occur in the time-windows prior to a given moment. The classification errors are considered based on the ensemble classification and not in each model classification output. For each time-window, a classifier is trained with the collection of documents, like in the baseline model, plus the previously introduced error collection and then stored. When a new collection of documents in the subsequent time-window occurs, all the previously trained classifiers are loaded, and will be classified as the newly seen examples participating equally to the final decision of the ensemble. Figure 1 depicts the proposed models.

### 4 Experimental Setup

#### 4.1 Dataset

The dataset we have defined to evaluate and validate our strategy includes 10 different *hashtags* that represent the different drifts, based on the assumption that they would denote mutually exclusive concepts, like *#real-madrid* and *#android*. By trying to use mutually exclusive concepts we intend to avoid misleading a classifier, as two different *tweets* could represent the same concept.

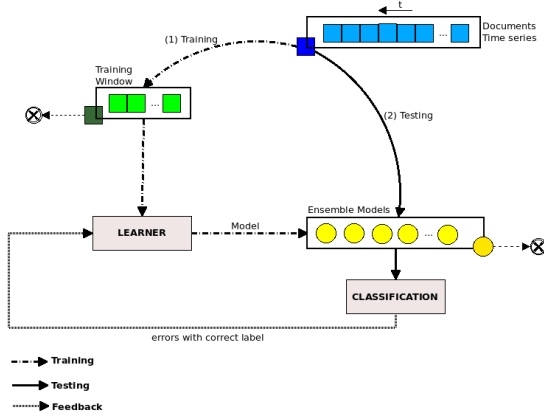


Figure 1: Proposed models

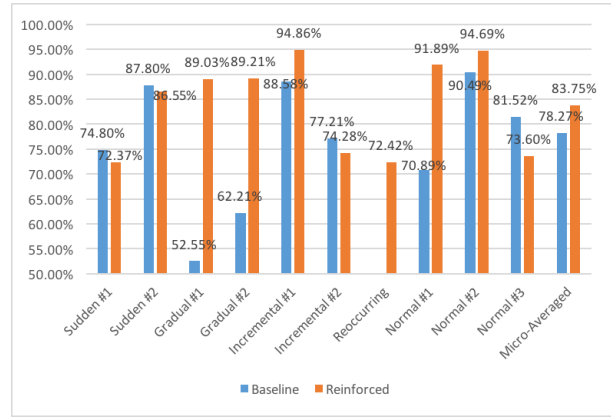


Figure 2: Micro-averaged F1

The Twitter API (`dev.Twitter.com`) was then used to request public *tweets* that contain the defined *hashtags*. The requests have been cared of between December 28, 2014 and January 21, 2015 and *tweets* were only considered if the user language was defined as English. *Tweets* containing no message content besides the *hashtag* were discarded. Finally, the *hashtag* was removed from the message content.

We have simulated the different types of drift by artificially defining timestamps to the previously gathered *tweets*. Time is represented as 100 continuous time windows, in which the frequency of each *hashtag* is altered in order to represent the defined drifts. Each *tweet* is then timestamped so it can belong to one of the time windows we have defined. A profound description of the used dataset can be found in [3]. Our final dataset contains 34.240 *tweets*.

## 4.2 Representation and Pre-processing

A *tweet* is represented as a vector space model, also known as *Bag of Words*. The collection of features is built as the dictionary of unique terms present in the documents collections. The *hashtag* was removed from the message content in order to be exclusively used as the document label.

High dimensional space can cause computational problems in text-classification problems where a vector with one element for each occurring term in the whole connection is used to represent a document. Also, over-fitting can easily occur which can prevent the classifier to generalize and thus the prediction ability becomes poor. Pre-processing methods were applied in order to reduce feature space. *Stopword removal* was then applied, preventing those non informative words from misleading the classification. *Stemming* method was also applied. Stemming does not alter significantly the information included, but it does avoid feature expansion.

## 4.3 Learning and Evaluation

The evaluation of our approach was done by the previously described dataset and using the Support Vector Machine (SVM). SVM was used in our experiments to construct the proposed models.

In order to evaluate the binary decision task of the proposed models we defined well-known measures based on the possible outcomes of the classification, such as, error rate ( $\frac{FP+FN}{TP+FP+TN+FN}$ ), recall ( $R = \frac{TP}{TP+FN}$ ), and precision ( $P = \frac{TP}{TP+FP}$ ), as well as combined measures, such as, the van Rijsbergen  $F_\beta$  measure, which combines recall and precision in a single score:  $F_\beta = \frac{(\beta^2+1)P \times R}{\beta^2 P + R}$ .  $F_\beta$  is mostly used in text classification problems with  $\beta = 1$ , i.e.  $F_1$ , an harmonic average between precision and recall.

## 5 Experimental Results and Analysis

We evaluate the performance obtained on the Twitter data set using the two approaches described in Section 3, namely the baseline model approach and the reinforced model approach. Figure 2 represents graphically the performance results obtained by classifying the dataset, considering the micro-averaged  $F_1$  measure. Analysing the graph we can observe that globally, and considering the average of the micro-averaged  $F_1$ , the storage of the priorly misclassified examples improves the overall classification. This is normal as the learning models are trained with more informative examples and this leads to a better performance. Most classes benefit from storing examples, and we have a significant improve in the average of the micro-averaged  $F_1$ , that increases from 78,27% to 83,27%,

but some classes, namely *Sudden#1*, *Sudden#2*, *Incremental#2* and *Normal#3* have a worst classification performance. We are confident that this decrease might be explained by the nature of the drift pattern.

As an example, a sudden drift is characterized by an abrupt increase of the frequency of a given class that occur during a period of time, followed by its disappearance. Storing examples that were misclassified, specially the positive ones that appeared firstly and remained misclassified until the classifier identified them as positive, will delude future classifiers, when the drift pattern is no longer represented. Although this is a supposition, that must be validated in future work, we also believe that it might be related to the class, that is the *hashtag* we have chosen to represent it. One of the possible problems that might arise from our approach is to store examples that are not representative of the class.

## 6 Conclusions and Future Work

We have proposed a method to determine the most relevant examples, by analysing their behaviour when defining separating planes or thresholds between classes. Those examples, deemed better than others, are kept for a longer time-window than the rest. The main idea is to boost the classification performance of learning models by providing additional and significant information.

The results revealed the usefulness of our strategy, as the results improved by 5% in comparing to the baseline approach, considering the average of the micro-averaged  $F_1$ . It is also important to conclude that we have shown that retaining informative examples can improve the learners' ability to identify a given class, independently from the drift pattern the class is representing. We do believe that it is problem dependent, even though it is an important insight in dynamic models, as they are particularly difficult learning scenarios. A special attention must be given to classes that tend to disappear, as retaining examples, in this particular case, for long periods can lead to misclassifications.

Our future work will include a more profound study about the longevity of those examples, i.e., for how long is it relevant to retain those examples.

## References

- [1] Indre Zliobaite. Learning under Concept Drift: an Overview. Tech. Report, Vilnius University, Faculty of Mathematics and Informatic, 2010.
- [2] Joana Costa, Catarina Silva, Mário Antunes, Bernardete Ribeiro. Concept Drift Awareness in Twitter Streams. In *Proc. of the 13th Int. Conference on Machine Learning and Applications*, pp. 294-299, 2014.
- [3] Joana Costa, Catarina Silva, Mário Antunes, and Bernardete Ribeiro. The Impact of Longstanding Messages in Micro-Blogging Classification. In *Proc. of the International Joint Conference on Neural Networks*, 2015.
- [4] M. Zappavigna. Ambient affiliation: A linguistic perspective on Twitter. In *New Media & Society*, vol. 13, no. 5, pp. 788-806, 2011.
- [5] A. Tsymbal. The problem of concept drift: definitions and related work. Dept Computer Science, Trinity College Dublin, Tech. Rep., 2004.
- [6] J. Kim, P. Bentley, U. Aickelin, J. Greensmith, G. Tedesco, J. Twycross. Immune system approaches to intrusion detection - a review. In *Natural Computing*, vol.6, no.4, pp. 413-466, 2007.
- [7] Joana Costa, Catarina Silva, Mário Antunes, and Bernardete Ribeiro. Defining Semantic Meta-Hashtags for Twitter Classification. In *Proc. of the 11th International Conference on Adaptive and Natural Computing Algorithms*, pp. 226-235, 2013.