ECAI 2010
H. Coelho et al. (Eds.)
IOS Press, 2010

doi:10.3233/978-1-60750-606-5-1141

GNUsmail: Open Framework for On-line Email Classification

José M. Carmona-Cejudo and **Manuel Baena-García** and **José del Campo-Ávila** and **Rafael Morales-Bueno** [1] and **Albert Bifet** [2]

Abstract. Real-time classification of massive email data is a challenging task that presents its own particular difficulties. Since email data presents an important temporal component, several problems arise: emails arrive continuously, and the criteria used to classify those emails can change, so the learning algorithms have to be able to deal with concept drift. Our problem is more general than spam detection, which has received much more attention in the literature.

In this paper we present GNUsmail, an open-source extensible framework for email classification, which structure supports incremental and on-line learning. This framework enables the incorporation of algorithms developed by other researchers, such as those included in WEKA and MOA. We evaluate this framework, characterized by two overlapping phases (pre-processing and learning), using the ENRON dataset, and we compare the results achieved by WEKA and MOA algorithms.

1 INTRODUCTION

The use of email grows every day, as a result of new applications for both personal and professional purposes. When emails are read and processed by the user, they can be easily stored if necessary. Thanks to the low cost of that storage, it is usual to file emails with a certain level of interest into some hierarchical category-based folders.

However, such advantages lead to an increasing volume of emails to be processed every day. The messages are usually stored into a specific place of the structure, which must be determined manually by the user. This can become a very time-consuming job. Although most email clients support hand-crafted rules for this task, these rules are usually static, inflexible, and considered to be too difficult to be used by most users [8]. Therefore, it would be desirable to automatically induce such rules.

Machine learning and *data mining* methodologies show how to induce *models* from a data set. Traditionally, batch methods have been applied to email classification, but recently on-line methods for dealing with new problems, such as concept drift or large volume of messages, are emerging. Email classification is a subfield of more general *text classification* research area, and its aim is to assign labels from a predefined set to each email. From a very simplified point of view, the following activities have to be carried out in order to construct a machine learning-based email classifier: *corpus preprocessing*, *model construction* and *validation*.

Little effort has been devoted to applying stream mining tools [7] to email classification. However, they are necessary when a lot of emails arrive continuously and must be processed, only once if possible (*incremental learning*), and also because folders are constantly created, moved or deleted, and the nature of emails within a folder may change over time (*concept drift*). To automatically produce *classifying models* taking into account all of the previous information, we need an on-line approach.

That said, the main goal of our contribution is the implementation of an *open framework for on-line email classification*. The proposed system allows the incorporation of stream-based algorithms, such as those included in the WEKA [11] and MOA [4] libraries, offering a flexible and extensible preprocessing module. The rest of this paper is organized as follows: in Section 2 our proposed preprocessing and learning framework is introduced, in Section 3 we explain the experimental evaluation, and in Section 4 we give our conclusions and mention some possibilities for future work.

2 GNUSMAIL DESIGN PHILOSOPHY

In this work we propose an extended version of GNUsmail [6], which incorporates incremental an on-line algorithms (including the management of concept drift) into the original architecture. GNUsmail is an open-source framework for adaptive email classification, with an extensible preprocessing module, based on the concept of 'filters' (each of these filters extract one or more attributes from raw email messages), and an equally extensible learning module (new algorithms, methods and libraries can be easily integrated).

Since arbitrary machine learning algorithms can be used inside our framework, GNUsmail is particularly useful for research purposes, as the framework does not limit itself to a single method or algorithm. The experimental evaluation of this work is an example, showing how new paradigms for classifiers are evaluated. Nevertheless, GNUsmail can also be used as a ready-to-use application, since an interface with the popular Mozilla Thunderbird email client has been implemented (see [6]). Our source code is licensed under the GPL license, and it is available at `http://code.google.com/p/gnusmail/`.

GNUsmail contains an email reader module, a text preprocessing module and a configurable learning module that can be configured to use specific algorithms and parameters. The preprocessing module deals with the extraction of attributes and their corresponding values from each email. When an incremental or on-line approach is followed, the email messages are read while the model is built. However, current models have an important limitation that affects this process: the learning attributes to be used to induce the model have to be fixed before beginning the induction of the algorithm since it is not possible to start to use a new attribute in the middle of the life-

[1] Universidad de Málaga, Spain, email: {jmcarmona, mbaena, jcampo, morales}@lcc.uma.es
[2] University of Waikato, New Zealand, email: abifet@cs.waikato.ac.nz

time of a learning model. Our main learning attributes are the most representative words in the email corpus, and thus some emails have to be read in advance, in order to decide which words are to be used as attributes. Hence, the first step of the training phase results in a list of attributes to be used by the model. After we have fixed these attributes, the incremental or on-line algorithm can start to analyze the incoming messages one by one, updating the model.

WEKA methods are used mainly with small datasets in environments without time and memory restrictions. MOA [5], on the other hand, is a data mining framework that adjusts to more demanding problems. In the streaming setting, data arrives at high speed and there are very strict time and space constraints.

We have used some ensemble methods from the MOA collection, such as `OzaBag` and `OzaBoost`, which outperform boosting in the streaming setting. For concept drift detection we have used DDM [10] and EDDM [1]. These methods detect change in the accuracy error of the classifier by comparing window statistics.

3 EXPERIMENTAL EVALUATION

We have evaluated our framework for both incremental and on-line learning schemes, using the well-known ENRON dataset [12]. We have used a *prequential* metric [9], that monitors the evolution of the system's performance over time.

For the evaluation of incremental methods in WEKA, we have used an updateable Naïve Bayes, IB-k and NN-ge models. The performance obtained by NN-ge is comparable with the best results obtained by [2] and [3], which use SVM-based approaches.

We have also evaluated the performance of several on-line algorithms with concept drift detection, using algorithms from the MOA framework. See Table 1 and Figure 1.

Table 1. Final folder-wise MOA prequential accuracies with bagging of NN-ge with DDM detector for some authors

User	Correct / Total	Percentage
farmer-d	2743/3590	76.41%
kaminsky-v	1798/2699	66.62%
lokay-m	1953/2479	78.78%
sanders-r	887/1207	73.49%
williams-w3	2653/2778	95.5%

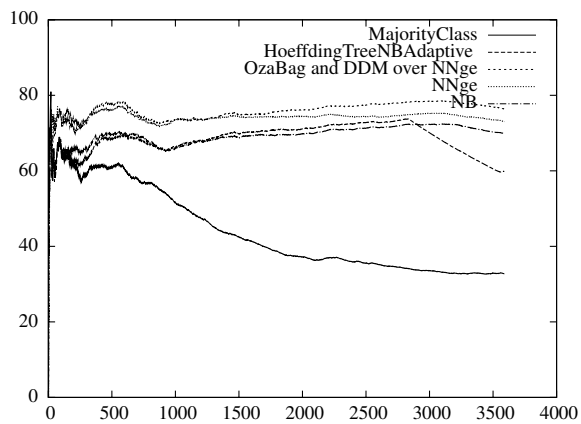

Figure 1. Evolution of the prequential error for farmer-d

4 CONCLUSION

In this paper we have presented GNUsmail, an open-source extensible framework for email classification which incorporates new methods for on-line learning with concept drift. GNUsmail flexible architecture allows to incorporate new filters for attribute generation, as well as alternative learning algorithms.

In our experiments we have compared MOA and Weka machine learning frameworks. We have found out that MOA outperforms the learning methods in Weka. By automatically adapting the model, concept-drift detection avoids a decline in performance when concept-drift occurs in the flow of messages.

With respect to on-line classifiers for email classification, some deficiencies have been detected. For example, it would be necessary to develop algorithms able to learn with only a few hundreds examples. Most importantly, up-to-date on-line methods need to know *a priori* all the attributes, values and classes. This limitation makes it necessary to process the data twice: first to extract features, and then to learn. Future methods should support on-line feature extraction, as well as learning.

Finally, considering feature extraction, it would be useful to include some new techniques in future work, such as named entity recognition, management of temporal features, or formality features, like spelling or the use of formal terms.

ACKNOWLEDGEMENTS

This work has been partially supported by the SESAAME project (code TIN2008-06582-C03-03) of the MEC, Spain.

REFERENCES

[1] Manuel Baena-García, José del Campo-Ávila, Raúl Fidalgo, Albert Bifet, Ricard Gavaldà, and Rafael Morales-Bueno, 'Early drift detection method', in *Fourth International Workshop on Knowledge Discovery from Data Streams*, (2006).

[2] R. Bekkerman, A. Mccallum, and G. Huang, 'Automatic categorization of email into folders: Benchmark experiments on Enron and SRI Corpora', Technical report, Center for Intelligent Information Retrieval, (2004).

[3] P. Bermejo, J. A. Gámez, J. M. Puerta, and R. Uribe-Paredes, 'Improving KNN-based e-mail classification into folders generating class-balanced datasets', in *Proceedings of the 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Sytems (IPMU-2008)*, pp. 529–536, (2008).

[4] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavaldà, 'New ensemble methods for evolving data streams', in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2009)*, pp. 139–148, (2009).

[5] Albert Bifet and Richard Kirkby. http://sourceforge.net/projects/moa-datastream/.

[6] J.M. Carmona-Cejudo, M. Baena-García, and R. R. Morales-Bueno, 'Amc-gnusmail: an extensible machine learning based framework for email classification', in *Proceedings of TTIA '09*. IASK, (2009).

[7] *Stream Data Management*, eds., N. Chaudhry, K. Shaw, and M. Abdelguerfi, Advances in Database Systems, Springer, 2005.

[8] E. Crawford, J. Kay, and E. McCreath, 'IEMS - the intelligent email sorter', in *Proceedings of the 19th International Conference on Machine Learning (ICML -2002)*, pp. 83–90, (2002).

[9] A. P. Dawid, 'Present position and potential developments: Some personal views: Statistical theory: The prequential approach', *Journal of the Royal Statistical Society. Series A*, **147**(2), 278–292, (1984).

[10] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, 'Learning with drift detection', in *SBIA Brazilian Symposium on Artificial Intelligence*, pp. 286–295, (2004).

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, 'The WEKA data mining software: An update', in *SIGKDD Explorations*, volume 11-1, pp. 10–18, (2009).

[12] B. Klimt and Y. Yang, 'The enron corpus: A new dataset for email classification research', in *Proceedings of the 15th European Conference on Machine Learning (ECML-2004)*, (2004).