ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

Semi-Supervised Classification of Social Textual Data Using WiSARD

Fabio Rangel[1], Fabrício Firmino[1], Priscila Machado Vieira Lima[1]
and Jonice Oliveira[1]

1- Federal University of Rio de Janeiro (UFRJ)
Pos-Graduation Program in Informatics (PPGI)
Av. Athos da S. Ramos, 149. Rio de Janeiro, RJ. - Brazil

Abstract.
Text categorization is a problem which can be addressed by a semi-supervised learning classifier, since the annotation process is costly and ponderous. The semi-supervised approach is also adequate in the context of social network text categorization, due to its adaptation to class distribution changes. This article presents a novel approach for semi-supervised learning based on WiSARD classifier (SSW), and compares it to other already established mechanisms (S3VM and NB-EM), over three different datasets. The novel approach showed to be up to fifty times faster than S3VM and EM-NB with competitive accuracies.

1 Introduction

In the field of Data Mining, text categorization constitutes a problem which is commonly tackled by supervised learning [1].Online Social Networks, however, present some extra challenges to the traditional approaches: (i) pieces of texts are very small; (ii) vocabulary is large; (iii) spelling is usually not corrected and (iv) new terms (slang and hashtags) appear frequently. Moreover, public opinion tends to change constantly, altering the distribution of data classes. As a consequence, the annotation step of a supervised classification approach would constitute a ponderous task. These conditions enhance the importance of using unlabeled data in the training step, learning improvements for further revision steps. Besides, semi-supervised learning is most useful whenever there is far more unlabeled than labeled data. This is likely to occur when obtaining data points is far cheaper (both computationally and financielly) than labelling them.[2].

WiSARD (Wilkie, Stonham & Aleksander's Recognition Device) [3] is a RAM based neural network. A WiSARD is composed of a set of individual classifiers, called discriminators, each one assigned to learn binary patterns belonging to a particular category. Therefore, a WiSARD has as many discriminators as the number of categories it should be able to distinguish [4]. In stream data mining context, WiSARD has the advantage of being a one shot classifier, allowing incremental online learning. We adapted the classifier to perform text categorization, using both labelled and unlabeled data.

In this paper, we compare our Semi-Supervised WiSARD (SSW) approach with other two approaches: Semi-Supervised Support Vector Machines (S3VM) [5] and Naive Bayes Expectation-Maximization [6]. The comparison was made over three different Datasets: Obama-McCain Debate (OMD) [7] and Stanford Twitter Sentiment Gold (STS-Gold) [8], both from Twitter, and Polarity Dataset v1.1 (IMDB) [9], from IMDB (Internet Movie Database). It was necessary to adapt the datasets to contain only two classes, because the S3VM model used as comparison was not able to perform multiclass categorization. We analyzed fitting time, predicting time, accuracy and standard deviation.

2 Semi-Supervised WiSARD (SSW)

We propose Semi-Supervised WiSARD (SSW) as a self training classifier based on WiSARD[3], which is a weightless neural network. WiSARD was first developed to recognize images, and the input pattern must be a binary feature vector, which we call *retina.* The RAM contents are modified by adding training patterns, so that incremental learning is performed. The training is supervised, which implies having to know the label of all patterns. Each category has a unique discriminator, which is composed by a set of RAMs. The RAM contents are initialized with zeros. Training updates the memories. Each RAM, maps a set of pseudo-randomized positions from the *retina*.

During the training, the input pattern bits define an addresses to access the RAMs. Each RAM updates the value in the position addressed. When predicting the class of a new pattern, WiSARD uses the same mapping to access the RAMs' addresses. If the address contains an integer higher than a predefined value (threshold), 1 is summed to the discriminator response. WiSARD decides the class of the predicting pattern by choosing the discriminator which returns the highest value. The threshold can be incremented in order to filter lower RAMs values in the case of a draw between two or more discriminators, or when these values are too close. This process is called *Bleaching* [10].

To adapt WiSARD to work on a Semi-Supervised fashion, we propose the use of a predicting confidence to decide which unlabeled pattern will be trained. The confidence metric is the same that has already been used on the Bleaching process: $c = 1 - r_1/r_2$. Where r_1 is the best result from a discriminator, and r_2 is the second best result. This way, an unlabeled pattern is trained only if the prediction confidence is greater than a predefined threshold. Text categorization domain required another adaptation of WiSARD due the sparsity of the feature vector. As many RAMs would access the zero position (those which all address bits are zeros), the prediction would happen based on absent features. To work around this issue, on full-zero patterns the memory does not contribute to the discriminator response, on SSW.

3 Metodology

3.1 Classifiers for Comparison

3.1.1 Semi-Supervised Support Vector Machines (S3VM)

The Semi-Supervised Support Vector Machine (S3VM) or Transductive SVM (TSVM) [5] is one of the popular semi-supervised classification algorithms that inherits the large-margin concept of SVMs [11]. In the transductive setting, the learner can observe the examples in the test set and potentially exploit the structure of their distribution [12]. A learning method is considered to be transductive if it only works on the labeled and unlabeled training data, not being able to deal with unseen data [13]. However, some approaches like TSVM [14] can be used as an inductive learning mechanism after the transductive learning phase, for the unseen data.

There are very few works focused on transforming SVM into a semi-supervised and *multiclass* approach [15]. S3VM model available was not suitable to perform *multiclass* classification. Its implementation is gradient-based [16] and the author provided it at his website[1].

3.1.2 Naive Bayes Expectation-Maximization (NB-EM)

Expectation-Maximization (EM) is a class of algorithms based on estimating the maximum likelihood (maximum a posteriori) iteratively [6]. It assumes that the documents are generated by a mixture of multinomials model, where each mixture component corresponds to a class [6]. One possible generating model for EM algorithms is Naive Bayes model. We used a Naive Bayes EM (NB-EM) implementation by Mathieu Blondel[2].

3.2 Datasets for Testing

Obama-McCain Debate (OMD) is a dataset composed by 3.238 tweets during the first day of USA television presidential debate, in September, 2008 [7]. Stanford Twitter Sentiment Gold (STS-Gold) is dataset which creation is based on a dataset called Stanford Twitter Sentiment (STS). STS contains 1.6 milion of tweets which labels were obtained thought *emoticons* [8]. STS-Gold is a sample of STS manually annotated with two labels: *positive* and *negative*. Polarity Dataset v1.1 (IMDB), which is a dataset of movie-reviews from Internet Movie Database, is introduced in [9].

3.3 Preprocessing

The preprocessing applied over the documents from all datasets followed 5 phases: (i) transforming all documents to lower case; (ii) removing the punctuations; (iii) removing links; (iv) removing mentions (”@” user name on Twitter);

[1] http://www.fabiangieseke.de/index.php/code/qns3vm

[2] https://gist.github.com/mblondel/f0789b921c98d0fe6868

Table 1: Number of features for each dataset after preprocessing and removal of classes, in order to leave only two classes.

Dataset	#Features	#Class 1	#Class 2
OMD	4848	1582	843
STS-Gold	4165	1401	632
IMDB	23151	700	700

(v) Stemming. The latter is a word standardization for text matching of morphological related terms. It consists of removing affixes, thus reducing the word to its stem [17]. The Porter Stemmer algorithm was applied at this step [18].

3.4 Experimental Design

The parameters for SSW (number of bits, boolean bleaching, confidence threshold and semi-supervised confidence) and S3VM (kernel function, regularization weight, unlabelled cost weight and sigma for RBF kernel) were obtained through the application of a genetic algorithm, having accuracy as goal function. EM-NB implementation had no parameters to optimize. In that genetic algorithm, for each tuple of parameters, we instantiated a classifier and calculated the accuracy via repeated random sub-sampling validation. In this validation, 20% was used for labeled training, 50% for unlabeled training and 30% for test. Since unlabeled data carries less information than labeled data, it is required, in large amounts of data, to significantly increase prediction accuracy [2]. After finding the best parameters, we repeated the validation 100 times to calculate the accuracy, standard deviation, average time to train and average time to predict. We included in the experiments a WiSARD classifier trained only on labelled data to be sure that semi-supervised approach was really improving the accuracy.

4 Experimental Results

The Table 2 shows the results using the three datasets. WiSARD classifier was added to the results, and SSW showed better accuracies. However, other WiSARD results are not being showed since it is not a semi-supervised approach and we are not comparing its results.

It is possible to see that S3VM had the best accuracy in all datasets. However, in two of these datasets S3VM's accuracy is only 2% better than SSW. In IMDB dataset, SSW did not show a competitive accuracy, but if we compare the fitting time, SSW fitted in 0.2 seconds, while S3VM needed 15 seconds. EM-NB approach showed a competitive accuracy at this dataset, but it needed more than one minute to fit. SSW also showed the best fitting time for all datasets. That constitutes an interesting result, since Data Stream Mining usually has time restrictions, demanding fast responses.

Table 2: Experimental results from 3 semi-supervised classifiers and 1 supervised classifier.

	SSW	S3VM	EM-NB	WiSARD
OMD				
Accuracy	0.6970	**0.7191**	0.6596	0.6866
Std	0.0171	0.0215	**0.0147**	0.0164
Fitting Time (s)	**0.0310**	0.4786	2.5181	-
Predicting Time (s)	0.0147	**0.0005**	0.3195	-
STS-Gold				
Accuracy	0.7597	**0.7781**	0.7108	0.7486
Std	**0.0180**	0.0243	0.0199	0.0185
Fitting Time (s)	**0.0291**	1.7577	2.2753	-
Predicting Time (s)	0.0128	**0.0005**	0.2646	-
IMDB				
Accuracy	0.6373	**0.6821**	0.6756	0.6208
Std	**0.0293**	0.0432	0.0426	0.0277
Fitting Time (s)	**0.2146**	15.1255	66.9370	-
Predicting Time (s)	0.1070	**0.0046**	1.7793	-

Another important point to consider is the standard deviation. S3VM and EM-NB had 67-68% of accuracy at IMDB dataset, but the standard deviation was 4.3%, while SSW had 2.9%. The predicting time for a SVM based approach is fast, but considering the sum of prediction and fitting time, SSW is the faster classifier. It is worth noting that EM-NB has no parameters to optimize, being an interesting semi-supervised classifier if one is working under time restriction, on a context different from Data Stream Mining.

5 Conclusion and Future Works

The present work presented a comparison between a novel self training based semi-supervised classifier (SSW), and established semi-supervised classifiers (S3-VM e EM-NB). SSW displayed the best fitting time and good standard deviations for all tested datasets. Fitting time is a important feature in the context of data stream mining, reinforcing the idea of using SSW. In accuracy comparison, S3VM showed the best accuracy but it took up to 15 seconds to fit one piece of data. SSW spent just 0.2 seconds, while still presenting a competitive accuracy.

For future works, we expect to test this novel approach in a Data Stream Mining larger dataset. In this context, it is important to develop a robust forgetfulness function for the classifier, in order to adapt to changes in the scenario.

References

[1] Charu C Aggarwal and ChengXiang Zhai. *Mining text data*. Springer Science & Business Media, 2012.

[2] Olivier Chapelle, Bernhard Schlkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 1st edition, 2010.

[3] Igor Aleksander, WV Thomas, and PA Bowden. Wisard a radical step forward in image recognition. *Sensor review*, 4(3):120–124, 1984.

[4] Mariacarla Staffa, Silvia Rossi, Maurizio Giordano, Massimo De Gregorio, and Bruno Siciliano. Segmentation performance in tracking deformable objects via wnns. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 2462–2467. IEEE, 2015.

[5] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *ICML*, volume 99, pages 200–209, 1999.

[6] Kamal Nigam, Andrew McCallum, and Tom Mitchell. Semi-supervised text classification using em. *Semi-Supervised Learning*, pages 33–56, 2006.

[7] David A. Shamma, Lyndon Kennedy, and Elizabeth F. Churchill. Tweet the debates: Understanding community annotation of uncollected sources. In *Proceedings of the First SIGMM Workshop on Social Media*, WSM '09, pages 3–10, New York, NY, USA, 2009. ACM.

[8] Hassan Saif, Miriam Fernandez, Yulan He, and Harith Alani. Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the sts-gold. 2013.

[9] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP*, pages 79–86, 2002.

[10] Bruno PA Grieco, Priscila MV Lima, Massimo De Gregorio, and Felipe MG França. Producing pattern examples from "mental" images. *Neurocomputing*, 73(7):1057–1064, 2010.

[11] Kohei Ogawa, Motoki Imamura, Ichiro Takeuchi, and Masashi Sugiyama. Infinitesimal annealing for training semi-supervised support vector machines. In *Proceedings of the 30th International Conference on Machine Learning*, pages 897–905, 2013.

[12] Thorsten Joachims et al. Transductive learning via spectral graph partitioning. In *ICML*, volume 3, pages 290–297, 2003.

[13] Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.

[14] Thorsten Joachims. Transductive support vector machines. *Chapelle et al.(2006)*, pages 105–118, 2006.

[15] Arkaitz Zubiaga, Víctor Fresno, and Raquel Martínez. Is unlabeled data suitable for multiclass svm-based web page classification? In *Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing*, SemiSupLearn '09, pages 28–36, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[16] Fabian Gieseke, Antti Airola, Tapio Pahikkala, and Oliver Kramer. Fast and simple gradient-based optimization for semi-supervised support vector machines. *Neurocomputing*, 123:23–32, 2014.

[17] Chris D Paice. An evaluation method for stemming algorithms. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 42–50. Springer-Verlag New York, Inc., 1994.

[18] Martin F Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.