

Four text classification algorithms compared on a Dutch corpus

Hein Ragas
Cap Gemini Netherlands
P.O. Box 2575,
3500 GN Utrecht, The Netherlands
HRagas@inetgate.capgemini.nl

Cornelis H.A. Koster
Department of Computer Science,
University of Nijmegen,
6525 ED Nijmegen, The Netherlands
kees@cs.kun.nl

Abstract We describe an experiment in applying text classification algorithms to Dutch texts. Four well-known learning algorithms, Rocchio's algorithm, the Simple Bayesian Classifier (SBC), the Sleeping Experts (SE) and Winnow were implemented. They were tested on a corpus of articles from the Dutch newspaper NRC, pre-classified into four categories. The algorithms are compared on learning speed and error rate. We also investigated the effect of discarding terms, using either a dynamic stoplist or the Winnow heuristic.

1 Introduction

For the DORO system [1] we needed a text classification algorithm suitable for many languages, including Dutch. We decided to implement from scratch a number of algorithms described in the literature and to perform a comparison of their effectiveness on Dutch texts. Some results from that comparison are presented here.

We have studied *disjoint classification* (each document is to be assigned to precisely one class) on the basis of keywords. As a measure of performance, the *speed of learning* is for us just as important as the final accuracy. Furthermore, we wanted to study the effect of stoplist strategies based on the terms encountered: later we shall use phrases as terms rather than keywords, so that using a static stoplist is out of the question.

2 The Experiment

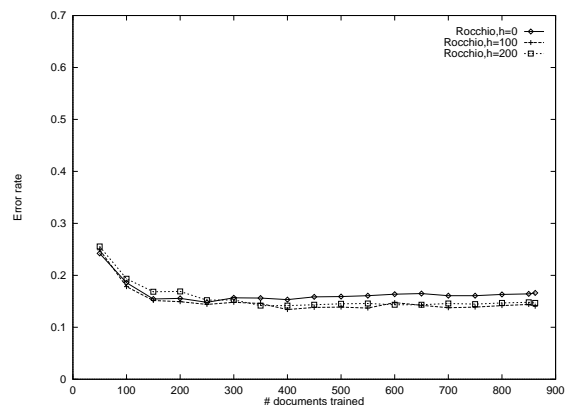
Our corpus was taken from the web pages of the Dutch newspaper NRC; for each publishing day the corpus contains four articles, the day's front page lead-article and one each on sports, economics and opinion. All available articles as of September '97 were downloaded and converted into plain ASCII-files, producing a corpus of 1436 documents, with 4 classes, with 359 documents in each class.

The NRC-corpus was randomized into three different orders of documents. Each of these three corpora was di-

vided into a *training* corpus and a *testing* corpus. 60% of the documents (862 documents) were taken as the training set. The rest of the documents (574 documents) were taken as the testing set. The classifiers were cumulatively trained in epochs of 50 documents. The data collected from the three experiments were averaged.

2.1 Rocchio

Rocchio's learning algorithm (see [2]) is in the classical IR tradition. It was originally designed to use relevance feedback in querying full-text databases, but it is easily adaptable to text classification. We investigated the use of a *dynamic stoplist*, discarding the h most frequent terms. The best results for Rocchio were obtained with $h = 100$, as can be seen in the following figure.

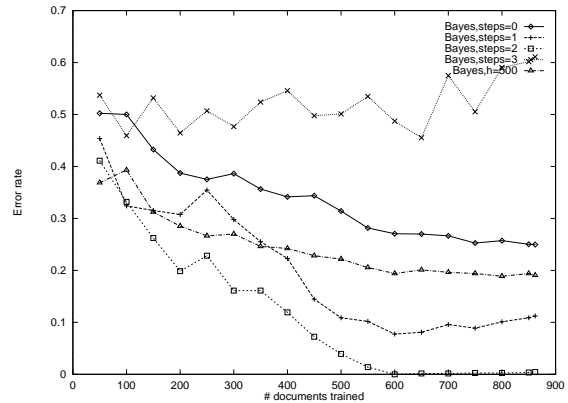
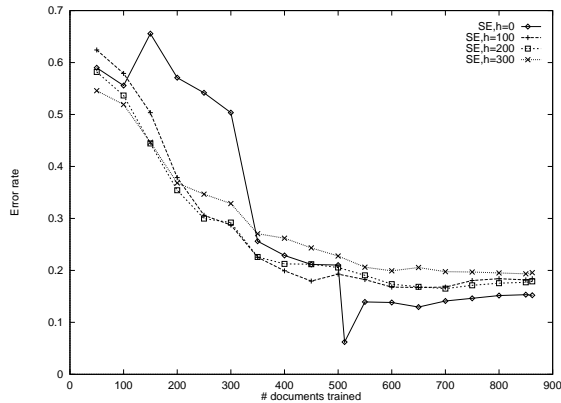


Most striking is the initial speed of training, and the lack of further learning after about 150 documents have been trained: the error rate stays at about 15%, with some minor fluctuations.

2.2 Sleeping Experts

The Sleeping Experts (SE) algorithm [2] learns by adjusting the weights of terms associated with classes each time that the term occurs in a document. The best result for Sleeping Experts was obtained with a cutoff at $h = 200$. The SE algorithm learns erratically and rather slowly; the final error rate is slightly worse than that of Rocchio.

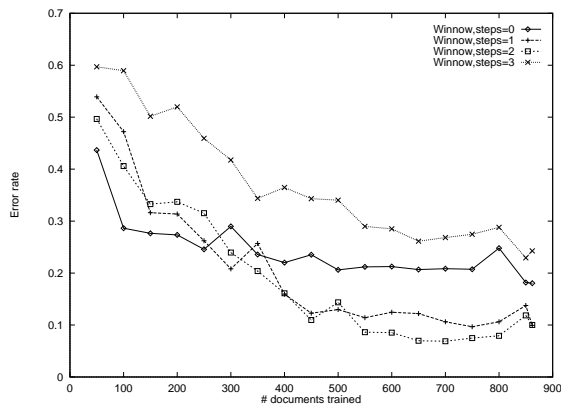
It should be pointed out that the SE algorithm on single keywords as terms is not at its best; it behaves much better using digrams, trigrams or even quadrigrams of words [2].



2.3 Winnow

The Winnow algorithm (actually Balanced Winnow [3]) is similar to the Sleeping Experts, but there is an important difference. Winnow is *mistake-driven*: it only modifies the weights of the terms in a class profile when the current weights would produce a mistake. Otherwise the weights are left unmodified. This process is iterated until the weights remain stable.

Dagan et al. [3] point out that the weighting scheme of their algorithm can also be used as a heuristic to limit the number of terms used in classification. All terms that have weights near to their initial values (having been demoted or promoted at most k steps away) have obviously little predictive value. Winnow itself performs best when discarding all terms within 1 or 2 Winnow-steps from their initial weight.



The graph shows a slow but steady effect of training. Ultimately, Winnow performs somewhat better than Rocchio.

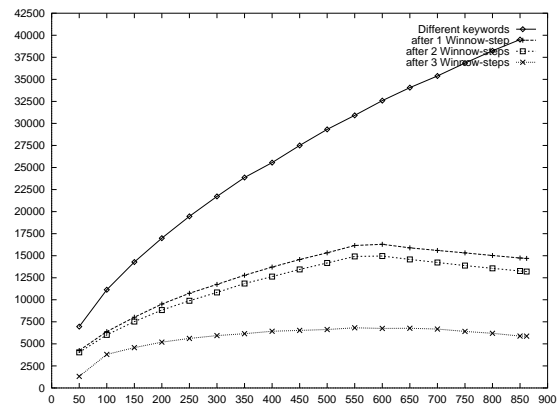
2.4 Simple Bayesian Classifier

The Simple Bayesian Classifier (SBC) [4] uses conditional probabilities to calculate the most probable class a document belongs to, given the document frequencies of terms in previous documents of each class.

Using a stop list with $h = 500$ had a positive effect on its behaviour, but the best results for the SBC were obtained by discarding all terms within 2 Winnow-steps from their original weight.

For SBC with 2 Winnow steps, the learning appears almost linear with the number of documents trained. The initial error-rate is worse than Rocchio's, but the SBC

keeps learning, until it makes almost no mistakes. After 600 documents trained – roughly the same size as the test corpus – its accuracy is spectacular. Evidently, the Winnow heuristic reduces the number of terms very effectively, as can be seen from the following graph.



3 Conclusions

The Winnow heuristic with 2 steps greatly improves the performance of SBC. Under the circumstances of our experiment, Rocchio and SBC performed better than Winnow, and a lot better than the Sleeping Experts. Rocchio learns faster than SBC, but it rapidly reaches a performance plateau.

A combination of Rocchio and SBC should give the best overall performance.

References

- [1] <http://www.profit.es/proyectos/dorocchios.html>
- [2] Cohen, W.W. and Singer, Y. (1996) Context-sensitive methods for text categorization. In: *Proceedings of SIGIR '96*.
- [3] Dagan, I., Karov, Y. and Roth, D. (1997) Mistake-Driven Learning in text Categorization. In: *Proceedings of the Second Conference on Empirical Methods in NLP*, p. 55-63.
- [4] Duda, R.O. and Hart, P.E. (1973) *Pattern classification and scene analysis* 1973, Wiley, New York.