A Probabilistic Approach to Document Classification

Bernard Merialdo
Institut Eurecom
BP 193
06904 Sophia-Antipolis
FRANCE
merialdo@eurecom.fr

Tel: +33 93 00 26 29
Fax: +33 93 00 26 27

A Probabilistic Approach to Document Classification

Abstract:

In this paper, we propose and experiment a probabilistic approach to document classification. We consider the problem of automatically assigning a new article to a Usenet newsgroup. To model a newsgroup, we build a probabilistic language model which is supposed to generate articles for this newsgroup. When a new article is presented, we use a Maximum A Posteriori rule to decide if the message was generated by this newsgroup or not. We evaluate this approach and compare it to a classification based on keywords. On these cases, the probabilistic approach gives better recall and precision indicators.

The paper is structured as follows: we first present the problem of document classification in general terms. We then describe our application to newsgroup classification and present the data that we are using. We present first results for a classification based on keyword selection. Finally, we describe the probabilistic formulation of the problem, experiment this approach on the same data and compare the results.

A Probabilistic Approach to Document Classification

Introduction

With the development of computer communications, a huge amount of information becomes accessible through the networks. New ways of communications have appeared, such as mailing lists and Usenet News (and more generally Bulletin Board Systems), where new information is constantly introduced. While these new media contain a lot of useful information (for a given user), they also contain a lot of information that is irrelevant to this user's motivations. In fact, many (most) of these sources have a low "signal-to-noise" ratio, and it is quite often that a new user, after a period of initial excitation, will stop using these because of the amount of time needed to access the specific pieces that fulfil his interest. These sources lack efficient Document Classification mechanisms, that will evaluate new information along a model of the user's interests, so that only information that is evaluated as "interesting" for the user will be presented [2].

One difficulty with these mechanisms is the way the users's model should be built. If building a new user's model requires some special skills from the user, or the intervention of some specialist, a wide acceptance of this mechanism will be difficult. Furthermore, as the user's interests evolve in time, maintaining and updating the model will be difficult.

Various approaches have been proposed to implement classification (also called filtering) systems. They range from rule-based methods [11], [12], where the rules are manually built by the user himself or by some operator who interacts with him, to statistically based methods [5], natural language understanding [13], automatic learning systems [10], collaborative voting [4] and even neural networks [14]. In this paper, we propose a probabilistic approach to Document Classification that automatically builds a user's model from previous responses of the user. We consider the experimental task of separating various Usenet newsgroups related to different aspects of Artificial Intelligence, i.e. we simulate a user who would be only interested in one of these aspects, and try to design a mechanism that will automatically detect the news article that deals with that aspect. The probabilistic approach assumes that the articles for each aspect are generated by a probabilistic language model, so that a simple bayesian rule can be used to classify a new article.We evaluate it and compare it to a classification based on keywords. The probabilistic approach performs better, in terms of recall and accuracy measurements, than the keyword approach.

Document Classification

The problem of Document Classification can be modelized as follows:

- there is a flow of incoming documents,
- a user looks at each document and assigns it to one among a set of predefined document classes.

For simplification, we consider that the user is interested in a single class only, so that the choice for a new document is simply to decide whether it belongs to that class or not. A Document Classification agent performs the same task and tries to obtain the same results as the user. Its performance is evaluated by comparing the agent's decisions with the assignment that the user would have provided.

More precisely, a new document can be considered as interesting (I) or uninteresting (U) by the user, and it can be selected (S) or discarded (D) by the agent. Thus there are 4 possibilities for an incoming document:

- interesting and selected IS,
- interesting and discarded ID,
- uninteresting and selected US,
- uninteresting and discarded UD

If the agent was matching the behaviour of the user perfectly, it would select only those documents that the user finds interesting and discard the others. In reality, there may be discrepancies, and some selected documents will be found uninteresting, while interesting documents may be discarded. The performance of an agent can be measured by the standard precision and recall rates:

- precision: the percentage of selected documents that are interesting,
- recall: percentage of interesting documents that are selected.

Those indicators are computed by the formulas:

$$p = \frac{N_{IS}}{N_{US} + N_{IS}} \qquad r = \frac{N_{IS}}{N_{ID} + N_{IS}}$$

The agent has to be customized for a particular user, and we need to get some information from him. We take the assumption that we use a minimum of information from the user, that is, just samples of evaluation of documents. There are several reasons for this assumption:

- we want to put a minimum of constraints on the user,
- we don't assume that the user understand the mechanism underlying the classification agent, or the effect of the parameters used to customize the agent,
- we expect that the focus of the user may slowly vary in time, so that the customization cannot be done once for all, but should be adapted continuously or at regular intervals,
- we don't want the customization to take user's or expert's time.

Asking a user to give a binary advice on the document is actually not the minimum that we could ask for. Some systems don't ask any information, and simply observe the behaviour of the user when browsing at documents. For example, the user may look at the titles and from that, decide if he wants to see more of the document. Articles that are not browsed are considered as uninteresting, while articles that are browsed are considered as interesting. This may lead to some false indication, since the user may be attracted by the title of an article, yet discover that the content of the article is not worth. This case cannot be detected by an agent which simply looks at the behaviour of the user, and that is why we prefer to ask directly the user whether the document was interesting or not. We feel that this binary indication can be provided by the user without unsupportable effort.

News Classification Application

We investigate the problem of classifying Usenet newsgroups. In order to easily construct experimental data, we chose to merge a number of newsgroups related to Artificial Intelligence (comp.ai, ai.edu, ai.fuzzy, ai.genetic, ai.nat-lang, ai.neural-nets, ai.nlang-know-rep, ai.philosophy, ai.shells, ai.vision). We then assume that a user is interested in a specific aspect of AI (corresponding to one of the original newsgroups) and try to identify it from the global flow. This mechanism allows to easily build a experimental setup without the requirement of user evaluation. The contents of the corresponding newsgroups were collected for a period of a few weeks, and separated into training data (used to build up the models) and test data (used to evaluate the performance of these models).

Keyword selection

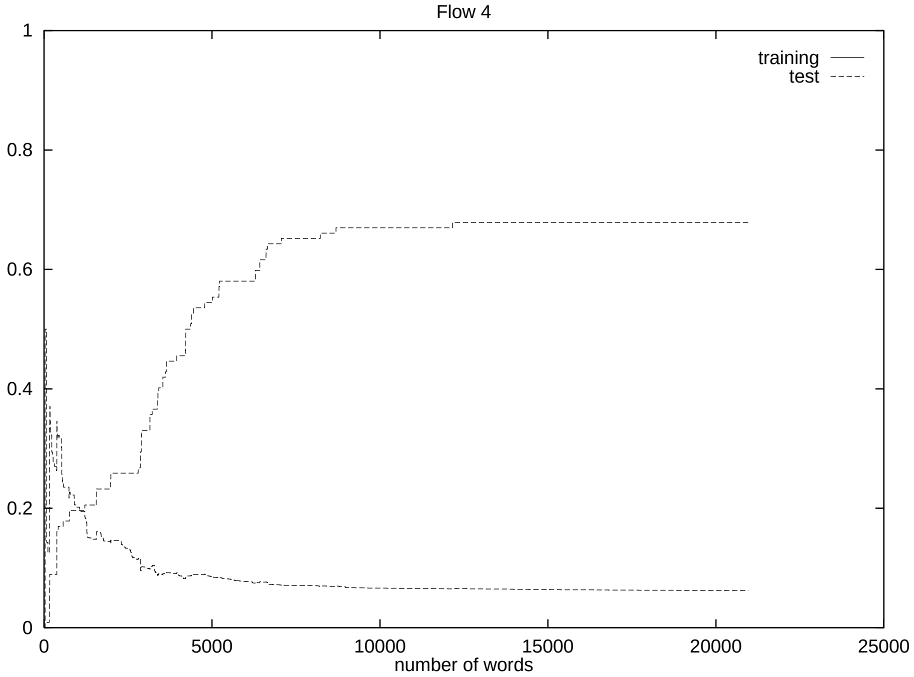
The first step is to select a set of keywords which are as representative as possible of each document flow. This selection is performed by computing the precision factor of each word with respect to each flow. If a word w appears in $n_i(w)$ documents of flow i (in the training data), then its precision factor is:

$$pr(w) = \frac{n_i(w)}{N(w)}$$

(N(w) is the total number of documents where the word w appears)

The meaning of the precision factor is simple: if we try to separate flow i and select documents if and only if they contain the keyword w, then the precision rate of the query is simply the precision factor of the keyword w for this flow of documents.

From this, we are able to construct a set of keyword-based queries by selecting the documents where at least one of the N-top keywords appear (this is the boolean OR of single-word queries). Following is a typical comparison of the recall (increasing) and precision (decreasing) rates of such queries when compared on training and test data.



Probabilistic approach

In the probabilistic formulation, we consider that the documents are generated by probabilistic sources, one for each class. Given a particular document D, the probability that it was generated by source S (rather than an other source) is:

$$p(S|D) = \frac{p(D|S) \cdot p(S)}{p(D)}$$

When the origin of a document is unknown, the best guess is to use a Maximum A Posteriori rule and to choose the source S that maximizes:

$$\mathbf{argmax} \quad p(S|D)$$

which can be also computed as:

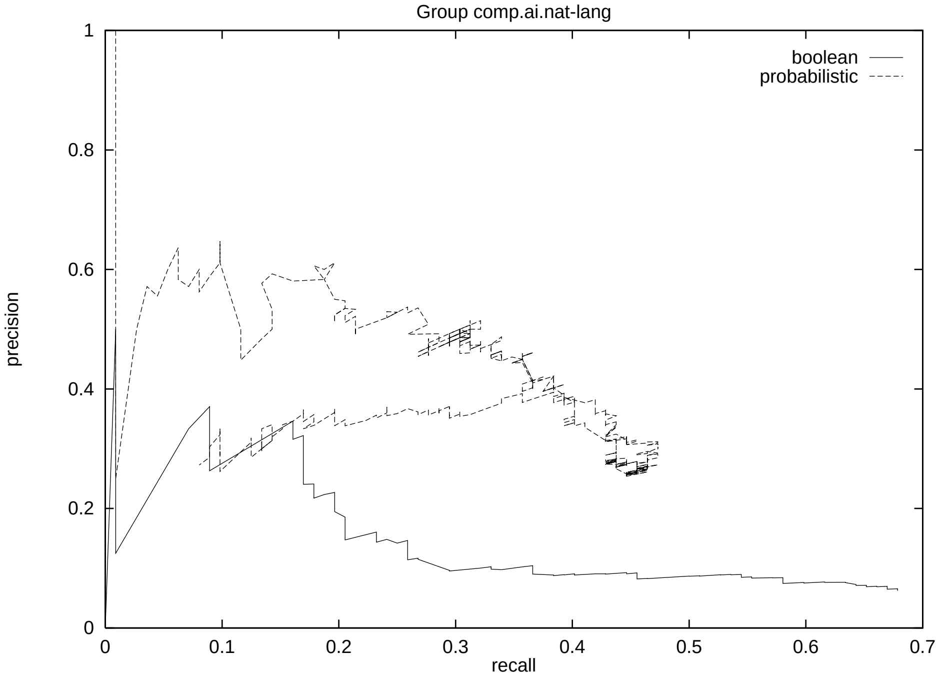
$$\mathbf{argmax} \quad p(D|S) \cdot p(S)$$

p(S) represents the a priori probability of the source, and p(D|S) the probability that the document is generated by the source. We approximate p(D|S) using a unigram model: the document D is considered as a sequence of words (including punctuation signs) $w_1, w_2, \ldots, w_n$, and we compute its probability of being generated by the source as:

$$p(D|S) = \prod_{i=1}^{n} p\left(w_i|S\right)$$

We calculate $p\left(w_i|S\right)$ by collecting the relative frequencies of the words on the training data. In practise, we restrict the vocabulary to a fixed set of words, and the words of the documents that are not in this set are replaced by an “unknown” sign. (Note that this formulation could as well use models based on bigrams, trigrams or more).

For our experiments, we build two language models: one for the newsgroup that is considered, the other for the rest of the messages. We assume both sources are equi-probable. A new document is classified based on the probabilities of being generated by each of these two sources. We computed the precision and recall rates for various sizes of vocabulary (taken as the N best keywords) and we plotted together the precision vs recall rates obtained by the probabilistic and the keyword-based (boolean) queries on test data:



For the keyword-based case, precision and recall rates are nearly monotonic function of the number of words that are considered in the request, because increasing the number of keywords considered automatically increases the number of documents that are selected. On the contrary, on the probabilistic case, adding new words leads to a new decision rule that may remove interesting documents that had been previously selected. This explains the more erratic behaviour of the recall rate for probabilistic queries. However, it can be noted that for a given recall rate, probabilistic queries have a better precision rate that boolean queries. This behaviour could be observed in all groups considered in the experiment.

Advantages of the probabilistic approach

We believe that the probabilistic approach has the following advantages:

- it is based on sound theoretical approach, where assumptions are made clearly,
- it comes with a wide variety of techniques, (hidden models, decision trees, classification...),
- adaptive mechanisms can be described easily,
- it was proven to be quite effective for tasks where knowledge modelling is difficult (speech recognition and language modelling).

The main drawbacks are probably that:

- it sometimes requires extensive computations,
- the (large) size of the training data is often a critical issue to build efficient systems.

Conclusion

We have described and experimented a probabilistic approach to Document Classification. It is based on a probabilistic language modelling of interesting and uninteresting documents, which allows to classify new documents using a bayesian rule. It was tested on the task of classifying Usenet articles coming from different sub-newsgroups related to Artificial Intelligence. When compared with a keyword-based approach, the probabilistic approach appeared to have better performance, in term of precision and recall.

Bibliography

[1] P. Baclace. “Competitive agents for information filtering.” *Communications of the ACM,*

35(12):50, Dec. 1992.

[2] N. Belkin and W. B. Croft. “Information filtering and information retrieval: two sides of the same coin?” *Communications of the ACM*, 35(12):29–38, Dec. 1992.

[3] S. Foltz, P.W.; Dumais. “Personalized information delivery: an analysis of information filtering methods.” *Communications of the ACM*, 35(12):51–60, Dec. 1992.

[4] D. Goldberg, D.and Nichols, B. Oki, and D. Terry. “Using collaborative filtering to weave an information tapestry.” *Communications of the ACM*, 35(12):61–70, Dec. 1992.

[5] P. S. Jacobs. “Using statistical methods to improve knowledge-based news categorization.” *IEEE Expert*, 8(2):13–23, Apr. 1993.

[6] K. Lai, T. Malone, and K. Yu. “Object lens: a spreadsheet for cooperative work.” *ACM Trans. Office Inf. Syst*, 6 (4):332–353, Apr. 1988.

[7] S. Loeb. “Architecting personalized delivery of multimedia information.” *Communications of the ACM*, 35(12):39–48, Dec. 1992.

[8] E. Lutz, H. von Kleist-Retzow, and K. Hoernig. *MAFIA-an active mail-filter-agent for an intelligent document processing support*, pages 235–251. Elsevier Science Publisher, 1990.

[9] W. MacKay, T. Malone, K. Crowston, R. Rao, D. Rosenblitt, and S. Card. “How do experienced information lens users use rules?” In *Proceedings of ACM CHI’89*, pages 211–216, 1989.

[10] P. Maes. “Agents that reduce work and information overload.” *Communications of the ACM*, 37(7):31–40, July 1994.

[11] Malone, T.W., Grant, K.R., and Turbak, F.A. “The information lens: An intelligent system for information sharing in organizations.” *Proceeding of the SIGCHI Human Factors in Computing Systems*, pages 1–8, 1986.

[12] S. Pollock. “A rule-based message filtering system.” *ACM Trans. Office Inf. Syst.*, 6(3):232–254, July 1988.

[13] A. Ram. “Natural language understanding for information-filtering systems.” *Communications of the ACM*, 35(12):39–48, Dec. 1992.

[14] J. C. Scholtes. “Neural nets for free-text information filtering.” In *Proceedings of the 3rd Australian Conference on Neural Nets, Canberra, Australia,*, Feb. 1992.