Experiments on the Use of Feature Selection and Negative Evidence in Automated Text Categorization

Luigi Galavotti[1], Fabrizio Sebastiani[2], and Maria Simi[3]

[1] AUTON S.R.L.
Via Jacopo Nardi, 2 – 50132 Firenze, Italy
galavott@tin.it
[2] Istituto di Elaborazione dell'Informazione – Consiglio Nazionale delle Ricerche
56100 Pisa, Italy
fabrizio@iei.pi.cnr.it
[3] Dipartimento di Informatica – Università di Pisa
56125 Pisa, Italy
simi@di.unipi.it

Abstract. We tackle two different problems of *text categorization* (TC), namely feature selection and classifier induction. *Feature selection* (FS) refers to the activity of selecting, from the set of r distinct features (i.e. words) occurring in the collection, the subset of $r' \ll r$ features that are most useful for compactly representing the meaning of the documents. We propose a novel FS technique, based on a simplified variant of the χ^2 statistics. *Classifier induction* refers instead to the problem of automatically building a text classifier by learning from a set of documents pre-classified under the categories of interest. We propose a novel variant, based on the exploitation of negative evidence, of the well-known k-NN method. We report the results of systematic experimentation of these two methods performed on the standard REUTERS-21578 benchmark.

1 Introduction

Text categorization (TC) denotes the activity of automatically building, by means of machine learning techniques, automatic text classifiers, i.e. systems capable of labelling natural language texts with thematic categories from a predefined set $C = \{c_1, \ldots, c_m\}$ (see e.g. [6]). In general, this is actually achieved by building m independent classifiers, each capable of deciding whether a given document d_j should or should not be classified under category c_i, for $i \in \{1, \ldots, m\}$[1]. This process requires the availability of a corpus $Co = \{d'_1, \ldots, d'_s\}$ of preclassified documents, i.e. documents such that for all $i \in \{1, \ldots, m\}$ and for all

[1] We here make the assumption that a document d_j can belong to zero, one or many of the categories in C; this assumption is verified in the REUTERS-21578 benchmark we use for our experiments. All the techniques we discuss here can be straightforwardly adapted to the other case in which each document belongs to exactly one category.

$j \in \{1, \ldots, s\}$ it is known whether $d'_j \in c_i$ or not. A general inductive process (called the *learner*) automatically builds a classifier for category c_i by learning the characteristics of c_i from a *training set* $Tr = \{d'_1, \ldots, d'_g\} \subset Co$ of documents. Once a classifier has been built, its effectiveness (i.e. its capability to take the right categorization decisions) may be tested by applying it to the *test set* $Te = \{d'_{g+1}, \ldots, d'_s\} = Co - Tr$ and checking the degree of correspondence between the decisions of the automatic classifier and those encoded in the corpus.

Two key steps in the construction of a text classifier are document indexing and classifier induction. *Document indexing* refers to the task of automatically constructing internal representations of the documents that (i) be amenable to interpretation by the classifier induction algorithm (and by the text classifier itself, once this has been built), and (ii) compactly capture the meaning of the documents. Usually, a text document is represented as a vector of weights $d_j = \langle w_{1j}, \ldots, w_{rj} \rangle$, where r is the number of features (i.e. words) that occur at least once in at least one document of Co, and $0 \leq w_{kj} \leq 1$ represents, loosely speaking, how much feature t_k contributes to the semantics of document d_j. Many classifier induction methods are computationally hard, and their computational cost is a function of r. It is thus of key importance to be able to work with vectors shorter than r, which is usually a number in the tens of thousands or more. For this, *feature selection* (FS) techniques are used to select, from the original set of r features, a subset of $r' \ll r$ features that are most useful for compactly representing the meaning of the documents. In this work we propose a novel technique for FS based on a simplified variant of the χ^2 statistics; we call this technique *simplified* χ^2. The key issues of FS and our simplified χ^2 method are introduced in Section 2, while the results of its extensive experimentation on REUTERS-21578, the standard benchmark of TC research, are described in Section 4.1.

Classifier induction refers instead to the inductive construction of a text classifier from a training set of documents that have already undergone indexing and FS. We propose a novel classifier induction technique based on a variant of k-NN, a popular instance-based method. After introducing instance-based methods in Section 3, in Section 3.1 we describe our modified version of k-NN, based on the exploitation of negative evidence. The results of its experimentation on REUTERS-21578 are described in Section 4.2. Section 5 concludes.

2 Issues in feature selection

Given a fixed $r' \ll r$, the aim of FS is to select, from the original set of r features that occur at least once in at least one document in Co, the r' features that, when used for document indexing, yield the best categorization effectiveness. The value $(1 - \frac{r'}{r})$ is called the *aggressivity* of the selection; the higher this value, the smaller the set resulting from FS, and the higher the computational benefits. On the other hand, a high agressivity may curtail the ability of the classifier to correctly "understand" the meaning of a document, since information that in

principle may contribute to specify document meaning is removed. Therefore, deciding on the best level of aggressivity usually requires some experimentation.

A widely used approach to FS is the so-called *filtering* approach, which consists in selecting the $r' \ll r$ features that score highest according to a function that measures the "importance" of the feature for the categorization task. In a thorough comparative experiment, performed across different classifier induction methods and different benchmarks, Yang and Pedersen [10] have shown

$$\chi^2(t_k, c_i) = \frac{g[P(t_k, c_i)P(\overline{t}_k, \overline{c_i}) - P(t_k, \overline{c_i})P(\overline{t}_k, c_i)]^2}{P(t_k)P(\overline{t}_k)P(c_i)P(\overline{c_i})} \quad (1)$$

to be one of the most effective functions for the filtering method, allowing aggressivity levels in the range [.90,.99] with no loss (or even with a small increase) of effectiveness. This contributes to explain the popularity of χ^2 as a FS technique in TC (see [6, Section 5]).

In Equation 1 and in those that follow, g indicates the cardinality of the training set, and probabilities are interpreted on an event space of documents (e.g. $P(\overline{t}_k, c_i)$ indicates the probability that, for a random document x, feature t_k does not occur in x and x belongs to category c_i), and are estimated by counting occurrences in the training set. Also, every function $f(t_k, c_i)$ discussed in this section evaluates the feature with respect to a specific category c_i; in order to assess the value of a feature t_k in a "global", category-independent sense, either the weighted average $f_{avg}(t_k) = \sum_{i=1}^{m} f(t_k, c_i)P(c_i)$ or the maximum $f_{max}(t_k) = \max_{i=1}^{m} f(t_k, c_i)$ of its category-specific values are usually computed.

In the experimental sciences χ^2 is used to measure how the results of an observation differ from those expected according to an initial hypothesis. In our application the initial hypothesis is that t_k and c_i are independent, and the truth of this hypothesis is "observed" on the training set. The features t_k with the lowest value for $\chi^2(t_k, c_i)$ are thus the most independent from c_i; as we are interested in those features which are not, we select those features for which $\chi^2(t_k, c_i)$ is highest.

However, Ng et al. [4] have observed that some aspects of χ^2 clash with the intuitions that underlie FS. In particular, they observe that the power of 2 at the numerator has the effect of equating the roles of the probabilities that indicate a positive correlation between t_k and c_i (i.e. $P(t_k, c_i)$ and $P(\overline{t_k}, \overline{c_i})$) and those that indicate a negative correlation (i.e. $P(t_k, \overline{c_i})$ and $P(\overline{t_k}, c_i)$). The function

$$CC(t_k, c_i) = \frac{\sqrt{g}[P(t_k, c_i)P(\overline{t}_k, \overline{c_i}) - P(t_k, \overline{c_i})P(\overline{t}_k, c_i)]}{\sqrt{P(t_k)P(\overline{t}_k)P(c_i)P(\overline{c_i})}} \quad (2)$$

they propose, being the square root of $\chi^2(t_k, c_i)$, emphasizes thus the former and de-emphasizes the latter. The experimental results by Ng et al. [4] show a superiority of $CC(t_k, c_i)$ over $\chi^2(t_k, c_i)$.

In this work we go a further step in this direction, by observing that in $CC(t_k, c_i)$, and *a fortiori* in $\chi^2(t_k, c_i)$:

- The $\sqrt{g}$ factor at the numerator is redundant, since it is equal for all pairs (t_k, c_i). This factor can thus be removed.

– The presence of $\sqrt{P(t_k)P(\overline{t}_k)}$ at the denominator emphasizes very rare features, since for these features it has very low values. By showing that document frequency is a very effective FS technique, [10] has shown that very rare features are the least effective in TC. This factor should thus be removed.
– The presence of $\sqrt{P(c_i)P(\overline{c}_i)}$ at the denominator emphasizes very rare categories, since for these categories this factor has very low values. Emphasizing very rare categories is counterintuitive, since this tends to depress microaveraged effectiveness (see Section 4), which is now considered the correct way to measure effectiveness in most applications [6, Section 8]. This factor should thus be removed.

Removing these three factors from $CC(t_k, c_i)$ yields

$$s\chi^2(t_k, c_i) = P(t_k, c_i)P(\overline{t}_k, \overline{c}_i) - P(t_k, \overline{c}_i)P(\overline{t}_k, c_i) \tag{3}$$

In Section 4 we discuss the experiments we have performed with $s\chi^2(t_k, c_i)$ on the REUTERS-21578 benchmark.

3 Issues in instance-based classifier induction

One of the most popular paradigms for the inductive construction of a classifier is the *instance-based* approach, which is well exemplified by the k-NN (for "k nearest neighbors") algorithm used e.g. by Yang [7]. For deciding whether d_j should be classified under c_i, k-NN selects the k training documents most similar to d_j. Those documents d'_z that belong to c_i are seen as carrying evidence towards the fact that d_j also belongs to c_i, and the amount of this evidence is proportional to the similarity between d'_z and d_j. Classifying a document with k-NN thus means computing

$$CSV_i(d_j) = \sum_{d'_z \in\ Tr_k(d_j)} RSV(d_j, d'_z) \cdot v_{iz} \tag{4}$$

where

– $CSV_i(d_j)$ (the *categorization status value* of document d_j for category c_i) measures the computed evidence that d_j belongs to c_i;
– $RSV(d_j, d'_z)$ (the *retrieval status value* of document d'_z with respect to document d_j) represents a measure of semantic relatedness between d_j and d'_z;
– $Tr_k(d_j)$ is the set of the k training documents d'_z with the highest $RSV(d_j, d'_z)$;
– the value of v_{iz} is given by

$$v_{iz} = \begin{cases} 1 \text{ if } d'_z \text{ is a positive instance of } c_i \\ 0 \text{ if } d'_z \text{ is a negative instance of } c_i \end{cases}$$

The threshold k, indicating how many top-ranked training documents have to be considered for computing $CSV_i(d_j)$, is usually determined experimentally on a validation set; Yang [7, 8] has found $30 \leq k \leq 45$ to yield the best effectiveness.

Usually, the construction of a classifier, instance-based or not, also involves the determination of a threshold τ_i such that $CSV_i(d_j) \geq \tau_i$ may be viewed as an indication to file d_j under c_i and $CSV_i(d_j) < \tau_i$ may be viewed as an indication not to file d_j under c_i. For determining this threshold we have used the *proportional thresholding* method, as in our experiments this has proven superior to *CSV thresholding* (see [6, Section 7]).

3.1 Using negative evidence in instance-based classification

The basic philosophy that underlies k-NN and all the instance-based algorithms used in the TC literature may be summarized by the following principle:

Principle 1 *If a training document d'_z similar to the test document d_j is a positive instance of category c_i, then use this fact as evidence towards the fact that d_j belongs to c_i. Else, if d'_z is a negative instance of c_i, do nothing.*

The first part of this principle is no doubt intuitive. Suppose d_j is a news article about Reinhold Messner's ascent of Mt. Annapurna, and d'_z is a very similar document, e.g. a news account of Anatoli Bukreev's expedition to Mt. Everest. It is quite intuitive that if d'_z is a positive instance of category Climbing, this information should carry evidence towards the fact that d_j too is a positive instance of Climbing. But the same example shows, in our opinion, that the second part of this principle is unintuitive, as the information that d'_z is a negative instance of category Fashion should not be discarded, but should carry evidence towards the fact that d_j too is a negative instance of Fashion.

In this work, we thus propose a variant of the k-NN approach in which *negative evidence* (i.e. evidence provided by negative training instances) is not discarded. This may be viewed as descending from a new principle:

Principle 2 *If a training document d'_z similar to the test document d_j is a positive instance of category c_i, then use this fact as evidence towards the fact that d_j belongs to c_i. Else, if d'_z is a negative instance of c_i, then use this fact as evidence towards the fact that d_j does not belong to c_i.*

Mathematically, this comes down to using

$$v_{iz} = \begin{cases} 1 \text{ if } d'_z \text{ is a positive instance of } c_i \\ -1 \text{ if } d'_z \text{ is a negative instance of } c_i \end{cases}$$

in Equation 4. We call the method deriving from this modification $k\text{-NN}^1_{neg}$ (this actually means $k\text{-NN}^p_{neg}$ for $p = 1$; the meaning of the p parameter will become clear later). This method brings instance-based learning closer to most other classifier induction methods, in which negative training instances play a fundamental role in the individuation of a "best" decision surface (i.e. classifier) that separates positive from negative instances. Even methods like Rocchio (see [6, Section 6]), in which negative instances had traditionally been either discarded or at best de-emphasized, have recently been shown to receive a performance boost by an appropriate use of negative instances [5].

4 Experimental results

In our experiments we have used the "REUTERS-21578, Distribution 1.0" corpus, as it is currently the most widely used benchmark in TC research[2]. REUTERS-21578 consists of a set of 12,902 news stories, partitioned (according to the "ModApté" split we have adopted) into a training set of 9,603 documents and a test set of 3,299 documents. The documents are labelled by 118 categories; the average number of categories per document is 1.08, ranging from a minimum of 0 to a maximum of 16. The number of positive instances per category ranges from a minimum of 1 to a maximum of 3964.

We have run our experiments on the set of 115 categories with at least 1 training instance, rather than on other subsets of it. The full set of 115 categories is "harder", since it includes categories with very few positive instances for which inducing reliable classifiers is obviously a haphazard task. This explains the smaller effectiveness values we have obtained with respect to experiments carried out by other researchers with exactly the same methods but on reduced REUTERS-21578 category sets (e.g. the experiments reported in [9] with standard k-NN). In all the experiments discussed in this section, stop words have been removed using the stop list provided in [3, pages 117–118]. Punctuation has been removed and letters have been converted to lowercase; no stemming and number removal have been performed. Term weighting has been done by means of the standard "ltc" variant of the $tf * idf$ function. Classification effectiveness has been measured in terms of the classic IR notions of precision (Pr) and recall (Re) adapted to the case of document categorization. We have evaluated *microaveraged* precision and recall, since it is almost universally preferred to *macroaveraging* [6, Section 8]. As a measure of effectiveness that combines the contributions of both Pr and Re, we have used the well-known function $F_1 = \frac{2 \cdot Pr \cdot Re}{Pr + Re}$. See the full paper for more details on the experiments.

4.1 Feature selection experiments

We have performed our FS experiments first with the standard k-NN classifier of Section 3 (with $k = 30$), and subsequently with a Rocchio classifier we have implemented following [1] (the Rocchio parameters were set to $\beta = 16$ and $\gamma = 4$; see [1, 5] for a full discussion of the Rocchio method). In these experiments we have compared two baseline FS functions, i.e. $\#_{avg}(t_k) = \sum_{i=1}^{m} \#(t_k, c_i)P(c_i)$ and $\chi^2_{max}(t_k) = \max_{i=1}^{m} \chi^2(t_k, c_i)$, to two variants of our $s\chi^2(t_k)$ function, i.e. $s\chi^2_{max}(t_k) = \max_{i=1}^{m} s\chi^2(t_k, c_i)$ and $s\chi^2_{avg}(t_k) = \sum_{i=1}^{m} s\chi^2(t_k, c_i)P(c_i)$. As a baseline, we have chosen $\chi^2_{max}(t_k)$ and not $\chi^2_{avg}(t_k)$ because the former is known to perform substantially better than the latter [10]. Table 1 lists the microaveraged F_1 values for k-NN and Rocchio with different FS techniques at different aggressivity levels. A few conclusions may be drawn from these results:

[2] The Reuters-21578 corpus may be freely downloaded for experimentation purposes from `http://www.research.att.com/~lewis/reuters21578.html`

Reduction level	k-NN				Rocchio			
	$\#(t_k)$	$\chi^2_{max}(t_k)$	$s\chi^2_{max}(t_k)$	$s\chi^2_{avg}(t_k)$	$\#(t_k)$	$\chi^2_{max}(t_k)$	$s\chi^2_{max}(t_k)$	$s\chi^2_{avg}(t_k)$
99.9	—	—	—	—	.458	.391	.494	—
99.5	—	—	—	—	.624	.479	.657	—
99.0	.671	.648	.697	.501	.656	.652	.692	—
98.0	.703	.720	**.734**	.554	.691	.710	.736	—
96.0	.721	.766	.729	.577	**.737**	**.733**	**.748**	—
94.0	.731	.766	.728	.596	—	—	—	—
92.0	.729	.772	.732	.607	—	—	—	—
90.0	.734	**.775**	.732	.620	—	—	—	—
85.0	**.735**	.767	.726	.640	—	—	—	—
80.0	.734	.757	.730	.658	—	—	—	—
70.0	.734	.748	.730	.682	—	—	—	—
60.0	.732	.741	.733	.691	—	—	—	—
50.0	.733	.735	.734	.701	—	—	—	—
40.0	.733	.735	.731	.716	—	—	—	—
30.0	.731	.732	.730	.721	—	—	—	—
20.0	.731	.732	.730	.727	—	—	—	—
10.0	.730	.730	.730	**.730**	—	—	—	—
00.0	.730	.730	.730	**.730**	—	—	—	—

Table 1. Microaveraged F_1 values for k-NN ($k = 30$) and Rocchio ($\alpha = 16$ and $\beta = 4$).

- on the k-NN tests we performed first, $s\chi^2_{avg}(t_k)$ proved largely inferior to $s\chi^2_{max}(t_k)$ (and to all other FS functions tested). This is reminiscent of Yang and Pedersen's [10] result, who showed that $\chi^2_{avg}(t_k)$ is outperformed by $\chi^2_{max}(t_k)$. As a consequence, due to time constraints we have abandoned $s\chi^2_{avg}(t_k)$ without further testing it on Rocchio;
- on the k-NN tests, $s\chi^2_{max}(t_k)$ is definitely inferior to $\chi^2_{max}(t_k)$ and comparable to $\#_{avg}(t_k)$ up to levels of reduction around .95, but becomes largely superior for aggressivity levels higher than that;
- following this observation, we have run Rocchio tests with extreme (from .960 up to .999) aggressivity levels, and observed that in these conditions $s\chi^2_{max}(t_k)$ outperforms both $\chi^2_{max}(t_k)$ and $\#_{avg}(t_k)$ by a wide margin.

The conclusion we may draw from these experiments is that $s\chi^2_{max}(t_k)$ is a superior alternative to both $\chi^2_{max}(t_k)$ and $\#_{avg}(t_k)$ when very aggressive FS is necessary. Besides, it is important to remark that $s\chi^2_{max}(t_k)$ is much easier to compute than $\chi^2_{max}(t_k)$. Altogether, these facts indicate that $s\chi^2_{max}(t_k)$ may be a very good choice in the context of learning algorithms that do not scale well to high dimensionalities of the feature space, such as neural networks, or in the application to TC tasks characterized by very high dimensionalities.

4.2 Classifier induction experiments

We have performed our classifier induction experiments by comparing a standard k-NN algorithm with our modified k-NN$^1_{neg}$ method, at different values of k. For

FS we have chosen $\chi^2_{max}(t_k)$ with .90 aggressivity since this had yielded the highest effectiveness ($F_1 = .775$) in the experiments of Section 4.1. The results of this experimentation are reported in the 1st and 2nd columns of Table 2.

	k-NN			k-NN$^1_{neg}$			k-NN$^2_{neg}$			k-NN$^3_{neg}$		
k	Re	Pr	F_1	Re	Pr	F_1	Re	Pr	F_1	Re	Pr	F_1
05	.711	.823	.763	.667	.821	.737	.709	.825	.764	.711	.823	.764
10	.718	.830	.770	.671	.918	**.775**	.720	.837	.774	.722	.834	.774
20	.722	.833	.774	.663	.930	.774	.725	.841	.780	.725	.836	.778
30	.714	.846	.775	.647	.931	.763	.722	.861	**.787**	.721	.854	**.782**
40	.722	.834	.774	.638	.934	.765	.731	.854	.786	.730	.841	.781
50	.724	.836	**.776**	.628	.938	.752	.730	.854	.786	.730	.843	.782
60	.724	.835	**.776**	.617	.940	.745	.730	.850	.785	.730	.842	.782
70	.722	.833	.774	.611	.945	.742	.731	.851	.786	.730	.842	.782

Table 2. Experimental comparison between k-NN and k-NN$^p_{neg}$ for different values of k and p, performed with χ^2_{max} FS and aggressivity .90, and evaluated by microaveraging.

A few observations may be made:

1. Bringing to bear negative evidence in the learning process has not brought about the performance improvement we had expected. In fact, the highest performance obtained for k-NN$^1_{neg}$ (.775) is practically the same as that obtained for k-NN (.776).
2. The performance of k-NN$^1_{neg}$ peaks at substantially lower values of k than for k-NN (10 vs. 50), i.e. much fewer training documents similar to the test document need to be examined for k-NN$^1_{neg}$ than for k-NN.
3. k-NN$^1_{neg}$ is a little less robust than k-NN with respect to the choice of k. In fact, for k-NN$^1_{neg}$ effectiveness degrades somehow for values of k higher than 10, while for k-NN it is hardly influenced by the value of k.

Observation 1 seems to suggest that negative evidence is not detrimental to the learning process, while Observation 2 indicates that, under certain conditions, it may actually be valuable. Instead, we interpret Observation 3 as indicating that negative evidence brought by training documents that are not very similar to the test document may be detrimental.

This is indeed intuitive. Suppose d_j is our news article about Reinhold Messner's ascent of Mt. Annapurna, and d'_z is a critical review of a Picasso exhibition. Should the information that d'_z is a negative instance of category c_i carry any evidence at all towards the fact that d_j too is a negative instance of c_i? Hardly so, given the wide semantic distance that separates the two texts. While very dissimilar documents have not much influence in k-NN, since positive instances are usually far less than negative ones, they do in k-NN$^1_{neg}$, since each of the k most similar documents, however semantically distant, brings a little weight to the final sum of which the CSV consists.

A similar observation lies at the heart of the use of "query zoning" techniques in the context of Rocchio classifiers [5]; here, the idea is that in learning a concept, the most interesting negative instances of this concept are "the least negative ones" (i.e. the negative instances most similar to the positive ones), in that they are more difficult to separate from the positive instances. Similarly, support vector machine classifiers [2] are induced by using just the negative instances closest to the decision surfaces (i.e. the so-called *negative support vectors*), while completely forgetting about the others.

A possible way to exploit this observation is switching to CSV functions that downplay the influence of the similarity value in the case of widely dissimilar documents; a possible class of such functions is

$$CSV_i(d_j) = \sum_{d'_z \in\ Tr_k(d_j)} RSV(d_j, d'_z)^p \cdot v_{iz} \tag{5}$$

in which the larger the value of the p parameter is, the more the influence of the similarity value is downplayed in the case of widely dissimilar documents. We call this method k-NN$^p_{neg}$. We have run an initial experiment, whose results are reported in the third and fourth column of Table 2 and which has confirmed our intuition: k-NN$^2_{neg}$ systematically outperforms not only k-NN$^1_{neg}$ but also standard k-NN.The k-NN$^2_{neg}$ method peaks for a higher value of k than k-NN$^1_{neg}$ and is remarkably more stable for higher values of k. This seemingly suggests that negative evidence provided by very dissimilar documents is indeed useful, provided its importance is de-emphasized. Instead, k-NN$^3_{neg}$ slightly underperforms k-NN$^2_{neg}$, showing that the level of de-emphatization must be chosen carefully.

5 Conclusion and further research

In this paper we have discussed two novel techniques for TC: $s\chi^2$, a FS technique based on a simplified version of χ^2, and k-NN$^p_{neg}$, a classifier learning method consisting of a variant, based on the exploitation of negative evidence, of the popular k-NN instance-based method.

Concerning the former method, in experiments performed on REUTERS-21578 simplified χ^2 has systematically outperformed χ^2, one of the most popular FS techniques, at very aggressive levels of reduction, and has done so by a wide margin. This fact, together with its low computational cost, make simplified χ^2 a very attractive method in those applications which demand radical reductions in the dimensionality of the feature space.

Concerning k-NN$^p_{neg}$, our hypothesis that evidence contributed by negative instances could provide an effectiveness boost for the TC task has been only partially confirmed by the experiments. In fact, our k-NN$^1_{neg}$ method has performed as well as the original k-NN but no better than it, and has furthermore shown to be more sensitive to the choice of k than the standard version. However, we have shown that by appropriately de-emphasizing the importance of very dissimilar training instances this method consistently outperforms standard k-NN. Given the prominent role played by k-NN in the TC literature, and given the simple

modification that moving from k-NN to k-NN$^p_{neg}$ requires, we think this is an interesting result.

References

[1] D. J. Ittner, D. D. Lewis, and D. D. Ahn. Text categorization of low quality images. In *Proceedings of SDAIR-95, 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 301–315, Las Vegas, US, 1995.

[2] T. Joachims. Text categorization with support vector machines: learning with many relevant features. In C. Nédellec and C. Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, number 1398 in Lecture Notes in Computer Science, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.

[3] D. D. Lewis. *Representation and learning in information retrieval*. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst, US, 1992.

[4] H. T. Ng, W. B. Goh, and K. L. Low. Feature selection, perceptron learning, and a usability case study for text categorization. In N. J. Belkin, A. D. Narasimhalu, and P. Willett, editors, *Proceedings of SIGIR-97, 20th ACM International Conference on Research and Development in Information Retrieval*, pages 67–73, Philadelphia, US, 1997. ACM Press, New York, US.

[5] R. E. Schapire, Y. Singer, and A. Singhal. Boosting and Rocchio applied to text filtering. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 215–223, Melbourne, AU, 1998. ACM Press, New York, US.

[6] F. Sebastiani. Machine learning in automated text categorisation: a survey. Technical Report IEI-B4-31-1999, Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, IT, 1999.

[7] Y. Yang. Expert network: effective and efficient learning from human decisions in text categorisation and retrieval. In W. B. Croft and C. J. van Rijsbergen, editors, *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 13–22, Dublin, IE, 1994. Springer Verlag, Heidelberg, DE.

[8] Y. Yang. An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1(1-2):69–90, 1999.

[9] Y. Yang and X. Liu. A re-examination of text categorization methods. In M. A. Hearst, F. Gey, and R. Tong, editors, *Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval*, pages 42–49, Berkeley, US, 1999. ACM Press, New York, US.

[10] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In D. H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 412–420, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US.