An Analysis of the Relative Difficulty of Reuters-21578 Subsets

Franca Debole, Fabrizio Sebastiani

Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche
Via Giuseppe Moruzzi, 1
56124 Pisa, Italy
{franca.debole,fabrizio.sebastiani}@isti.cnr.it

Abstract

The existence, public availability, and widespread acceptance of a standard benchmark for a given information retrieval (IR) task are beneficial to research on this task, since they allow different researchers to experimentally compare their own systems by comparing the results they have obtained on this benchmark. The Reuters-21578 test collection, together with its earlier variants, has been such a standard benchmark for the text categorization (TC) task throughout the last ten years. However, the benefits that this has brought about have somehow been limited by the fact that different researchers have "carved" different subsets out of this collection, and tested their systems on one of these subsets only; systems that have been tested on different Reuters-21578 subsets are thus not readily comparable. In this paper we present a systematic, comparative experimental study of the three subsets of Reuters-21578 that have been most popular among TC researchers. The results we obtain allow us to determine the relative difficulty of these subsets, thus establishing an indirect means for comparing TC systems that have, or will be, tested on these different subsets.

1. Introduction

The existence, public availability, and widespread acceptance of a standard benchmark for a given information retrieval (IR) task are beneficial to research on this task, since they allow different researchers to experimentally compare their own systems by comparing the results they have obtained on this benchmark.

The Reuters-21578 test collection, together with its earlier variants, has been such a standard benchmark for the text categorization (TC) task (Sebastiani, 2002) throughout the last ten years. Reuters-21578 is a set of 21,578 news stories appeared in the Reuters newswire in 1987, which are classified according to 135 thematic categories, mostly concerning business and economy. This collection has several characteristics that make it interesting for TC experimentation:

- similarly to many other applicative contexts, it is multi-label, i.e. each document d_i may belong to more than one category;
- the set of categories is not exhaustive, i.e. some documents belong to no category at all;
- the distribution of the documents across the categories is highly skewed, in the sense that some categories have very few documents classified under them ("positive examples") while others have thousands;
- there are several semantic relations among the categories (e.g. there is a category WHEAT and a category GRAIN, which are obviously related), but these relations are "hidden" (i.e. there is no explicit hierarchy defined on the categories).

This collection is also fairly challenging for TC systems based on machine learning (ML) techniques, since several categories have (under any possible split between training and test documents) very few training examples, making the inductive construction of a classifier a hard task. All of these properties have made Reuters-21578 the benchmark of choice for TC research in the past years.

Unfortunately, the benefits to TC research that Reuters-21578 has brought about have been somehow limited by the fact that different researchers have "carved" different subcollections out of this collection, and tested their systems on one of these subcollections only. The most frequent direction for extracting a subcollection out of Reuters-21578 has been that of restricting the attention to a subset of categories only. The subsets that have been most frequently used in TC experimentation are[1]:

- the set of the 10 categories with the highest number of positive training examples (hereafter, R(10));
- the set of the 90 categories with at least one positive training example and one positive test example (hereafter, R(90));
- the set of the 115 categories with at least one training example (hereafter, R(115)).

Systems that have been tested on these different Reuters-21578 subsets are thus not readily comparable. In this paper we present a systematic, comparative experimental study of the above-mentioned three subsets of Reuters-21578. We test the relative difficulty of these subsets in a variety of experimental TC contexts, generated by two different term weighting policies, three different feature selection functions, three different "reduction factors" for feature selection, three different learning methods, and two

[1] As for which Reuters-21578 documents are used as training examples, we here refer to the "ModApté split", a partition of the collection into a training set and a test set that has almost universally been adopted by TC experimenters. See Section 2. for more details.

different experimental measures, in all possible combinations. Our results allow us to obtain a reliable estimation of the relative difficulty of these subsets, thus establishing an indirect means for comparing TC systems that have, or will be, tested on these different subsets.

This paper is structured as follows. In Section 2. we describe in some detail the Reuters-21578 test collection and the subsets of it that have been used most often in TC research. Section 3. presents a systematic experimental study in which we test the relative difficulty of these subsets and give theoretical justifications for these results. Section 4. concludes.

2. The Reuters-21578 collection and its subsets

The data contained in the "Reuters-21578, Distribution 1.0" corpus consist of 21,578 news stories appeared on the Reuters newswire in 1987[2]. The Reuters-21578 documents actually used in TC experiments are only 12,902, since the creators of the collection found ample evidence that the other 8,676 documents had not been considered for labelling by the people who manually assigned categories to documents ("indexers"). In order to make different experimental results comparable, standard "splits" (i.e. partitions into a training and a test set) have been defined by the creators of the collection on the 12,902 documents. Apart from very few exceptions, TC researchers have used the "ModApté" split, in which 9,603 documents are selected for training and the other 3,299 form the test set. In this paper we will always refer to the ModApté split.

The TOPICS group of categories contains 135 categories. Among them, 20 have (in the ModApté split) no positive training documents; as a consequence, these categories have never been considered in any TC experiment, since the TC methodology requires deriving a classifier either by automatically training an inductive method on the training set only, and/or by human knowledge engineering based on the analysis of the training set only.

Since the 115 remaining categories have at least one positive training example each, in principle they can all be used in experiments. However, several researchers have preferred to carry out their experiments on different subsets of categories. Globally, the three subsets that have been most popular are

- The set of the 10 categories with the highest number of positive training examples (hereafter, R(10)).
- The set of 90 categories with at least one positive training example and one test example (hereafter, R(90)). This appears to be the most frequently chosen subset.
- The set of 115 categories with at least one positive training example (R(115)).

It follows from this discussion that R(10) $\subset$ R(90) $\subset$ R(115).

[2]The Reuters-21578 corpus is freely available for experimentation purposes from `http://www.daviddlewis.com/resources/`

Reasons for using one or the other subset have been different. The only clear fact is that the 10 most frequent categories provide an easier testbed than the other two sets, although it is not clear exactly *how easier*. Furthermore, it is not clear at all whether R(90) is any easier than R(115). The experiments that we describe in this section are exactly aimed at answering these two questions, and in general at establishing the relative difficulty of the three relevant Reuters-21578 subsets.

3. Experiments

The experiments we have conducted test the relative difficulty of the three above-mentioned Reuters-21578 subsets in *all* experimental TC contexts corresponding to any combination of a learning method, a term selection function, a reduction factor, a term weighting policy, and an effectiveness function, chosen from the following.

- As for the *learning methods*, we have used a choice among (i) a standard Rocchio method for learning linear classifiers, (ii) a standard k-NN algorithm, and (iii) the support vector machine (SVM) learner as implemented in the SVMLIGHT package (version 3.5) (Joachims, 1999). For reasons of brevity we do not discuss these methods in detail; the interested reader will find detailed presentations of them in (Debole and Sebastiani, 2003a).
- As for the *term selection functions*, we have used a choice among the three functions $\{\chi^2, IG, GR\}$ (see (Debole and Sebastiani, 2003a) for their mathematical form). The first two (chi-square and information gain) are standard tools-of-the-trade in the term selection literature, while the third is an entropy-normalized version of information gain whose use as a term selection function was first proposed in (Debole and Sebastiani, 2003b). Each of the three functions has been used according to the global policy (see [Section 5.3](Sebastiani, 2002)), essentially for efficiency reasons. Globalization has been achieved by means of the f_{max} function, the globalization function of choice in the TC literature, defined as $f_{max}(t_k) = \max_{i=1}^{|\mathcal{C}|} f(t_k, c_i)$.
- As for the *reduction factors* for feature selection, we have used a choice among the three values $\xi \in \{0.90, 0.50, 0.0\}$, where a 0.0 reduction factor means no reduction at all.
- As for the *term weighting policies*, we have used a choice between a standard, cosine-normalized form of $tf*idf$, or a *supervised term weighting* policy (Debole and Sebastiani, 2003b), consisting in replacing the idf component of $tf * idf$ with the function that, in the same experiment, has been previously used for term selection (this yields e.g. cosine-normalized $tf * GR$ if GR has been previously used for feature selection).
- As for the *effectiveness functions*, we have considered both the microaveraged and macroaveraged version of the F_1 function.

	Microaveraged F_1		Macroaveraged F_1	
	Avg	**StDev**	**Avg**	**StDev**
R(10)	0.852	0.048	0.715	0.097
R(90)	0.787	0.059	0.468	0.068
R(115)	0.784	0.062	0.494	0.118

Table 1: Average effectiveness and standard deviation scores averaged across all the text classifiers tested in our experiments on the three Reuters-21578 subsets.

In all the experiments discussed in this paper, stop words have been removed using the stop list provided in (Lewis, 1992, pages 117–118), punctuation has been removed, all letters have been converted to lowercase, numbers have been removed, and stemming has been performed by means of Porter's stemmer.

3.1. Experimental results

For reasons of space the detailed results of our experiments are omitted; the interested reader can consult (Debole and Sebastiani, 2003a). Figure 1 summarizes these results by averaging them for each studied technique. For instance, the curve marked "SVM" reports the average results of all the experiments run with the SVM learner. This means that the average is computed across all possible combinations of term weighting policies, feature selection policies, feature selection functions, and reduction factors for feature selection; separate plots for microaveraged F_1 and macroaveraged F_1 are given. Table 1 reports mean and standard deviation scores obtained across *all* 48 different experiments, and can thus be considered fairly representative. Finally, Table 2 reinterprets the results of Table 1 in terms of relative difficulty of the three Reuters-21578 subsets studied; the values contained in the table can be used for computing the likely performance that a given method tested on Reuters-21578 subset x could approximately have obtained if tested on subset y.

The fact that emerges most clearly from these experiments is that R(10) is the easiest subset, regardless of the choice of learning method, feature selection function, effectiveness function, etc. This was largely to be expected, given that its categories are the ones with the highest number of positive examples, and as such allow taming the "curse of dimensionality" more effectively.

On average, the decrease in performance in going from R(10) to R(90) is much sharper for macroaveraging (-53.1%) than for microaveraging (-7.6%). This can be explained by the fact that microaveraged effectiveness is dominated by the performance of the classifiers on the most frequent categories. To see this, note that microaveraged F_1 is an increasing function of microaveraged precision and microaveraged recall, and that:

- Microaveraged recall is the proportion of correct positive classification decisions that are indeed taken, and most correct positive classification decisions by definition concern categories that have many positive test examples. In Reuters-21578 the 10 categories that have the highest number of positive *test* examples are (unsurprisingly, given that the train/test partition was obtained by a random split) the same categories that have the highest number of positive *training* examples, i.e. are the categories in R(10). Note that the 10 categories in R(10) have altogether 2787 test examples, while the other 80 categories in R(90) have altogether just 957 of them; this shows that the former set of categories contributes three times as much as the latter in determining microaveraged recall on R(90).
- Microaveraged precision is the proportion of the positive classification decisions taken that are indeed correct, and it can be expected that most positive classification decisions taken concern categories that have many positive test examples, which are, as noted above, the same categories that have many positive *training* examples.

As a result, the microaveraged performance obtained on R(90) is heavily influenced by the performance obtained on the 10 most frequent categories, and much less heavily by the performance obtained on the remaining 80 categories. This explains why the above-mentioned decrease in microaveraged effectiveness is not very sharp. Instead, macroaveraged effectiveness is, by definition, not dominated by any category in particular. Since each of the 80 least frequent categories counts the same as any of the 10 most frequent ones, the fact that the former categories are more difficult than the latter weighs heavily on macroaveraged effectiveness, and the decrease in performance is more marked.

A second fact that also emerges clearly from the experiments is that R(115) is not significantly harder than R(90) when effectiveness is computed through microaveraging (-0.3%), while it is even easier (+5.5%) if macroaveraging is used. Both facts seem, on the surface, surprising, since the 25 additional categories have on average much fewer training examples (2.52 each) than the other 90 (107 each). However, arguments similar to the ones expoused above show that there is indeed a rationale for this. Microaveraged effectiveness is marginally hurt by the performance obtained on the 25 additional categories, since these categories contain no positive test examples: this means that microaveraged recall is by definition unaffected, while microaveraged precision is (for the same reasons discussed below re: macroaveraged precision) hurt only scarcely.

The fact that macroaveraged effectiveness even *benefits* from the added 25 categories is less obvious, but can be explained by the following fact. The value of F_{1_i} is equal to 1 for each category c_i on which no negative test examples are incorrectly classified under c_i (it is 0 otherwise). In order for this to happen, the threshold τ_i needs to be set high enough that for no test document d_j the CSV will exceed it. This indeed happens frequently, since the validation set on which τ_i is tuned also contains very few positive examples (if any – these 25 categories have, on average, 2.52 training *or* validation examples); this means that, in order to correctly classify the validation examples, high values for τ_i tend to be chosen.

A fact that emerges clearly from the low values of standard deviation reported in Table 1 is that these conclusions

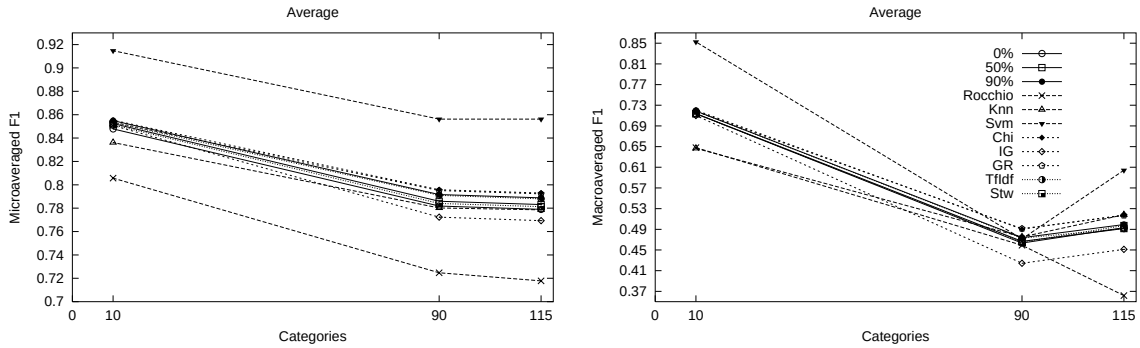

Figure 1: Plots of micro-averaged F_1 (leftmost) and macro-averaged F_1 (rightmost) obtained by averaging across term weighting policies, feature selection policies, feature selection functions, reduction factors for feature selection, and learning methods. The X axis indicates the three subsets of Reuters-21578 described in Section 2..

	Microaveraging			Macroaveraging		
	R(10)	R(90)	R(115)	R(10)	R(90)	R(115)
R(10)	–	+8.2%	+8.6%	–	+46.8%	+44.6%
R(90)	-7.6%	–	+0.3%	-53.1%	–	-5.2%
R(115)	-7.9%	-0.3%	–	-50.5%	+5.5%	–

Table 2: Values of relative difficulty of Reuters-21578 subsets as derived from the average effectiveness values of Table 1. The value in a given entry measures how easier the subset in the row proved with respect to the subset in the column.

are largely independent of the techniques employed, regardless of whether they are concerned with learning, or feature selection, or weighting, etc. Figure 1 tells us that, while for macroaveraging some exceptions to the general trend do exist (e.g. the Rocchio learner performs worse on R(115) than on R(90)), microaveraging displays little or no variance across different techniques. This suggests that our conclusion are fairly reliable, even if this degree of reliability cannot formally be measured.

4. Conclusion

We have presented a systematic, comparative experimental study of the three most popular subsets of Reuters-21578, itself the most popular test collection of text categorization research. We have carried out experiments on a variety of experimental contexts, including all possible combinations of three learning methods, three term selection functions, three term selection reduction factors, two term weighting policies, and two effectiveness functions. The results we have obtained are thus fairly representative of the relative difficulty of the three Reuters-21578 subsets, also as a result of the fact that the design choices that we have tested are widely different among each other and, at the same time, widely used in the text categorization literature. We have also presented theoretical, *a posteriori* justifications for these results, in particular explaining (i) why the decrease in performance that can be expected in going from R(10) to R(90) is sharper for macroaveraging than for microaveraging, and (ii) why in going from R(90) to R(115) we may expect almost no decrease in microaveraged performance, and even an increase in macroaveraged performance.

The cumulative results we have obtained, which are conveniently summarized in Table 2, finally allow the comparison, albeit indirect, of different text classifiers which, in individual experiments, had been or will be tested by their proponents on different Reuters-21578 subsets.

5. References

Debole, Franca and Fabrizio Sebastiani, 2003a. An analysis of the relative hardness of reuters-21578 subsets. Technical report, 2003-TR-49,Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, IT. Submitted for publication.

Debole, Franca and Fabrizio Sebastiani, 2003b. Supervised term weighting for automated text categorization. In *In Proceedings of SAC-03, 18th ACM Symposium on Applied Computing*. Melbourne,US: ACM Press.

Joachims, Thorsten, 1999. Making large-scale SVM learning practical. In Bernhard Schölkopf, Christopher J. Burges, and Alexander J. Smola (eds.), *Advances in Kernel Methods – Support Vector Learning*, chapter 11. Cambridge, US: The MIT Press, pages 169–184.

Lewis, David D., 1992. *Representation and learning in information retrieval*. Ph.D. thesis, Department of Computer Science, University of Massachusetts, Amherst US.

Sebastiani, Fabrizio, 2002. Machine learning in automated text categorization. *ACM Computing Surveys*:34(1)1–47.