

The Bootstrapping Problem

C.H.A. Koster*, P. Jones†, M. Vogel†, and N.Gietema‡

OTC Workshop, SIGIR 2002

1 Introduction

Once upon a time there was a company, which made a living by collecting documents in a welldefined area, classifying them by subject, assigning keywords and presenting them to an audience of professionals working in this area in the form of a specialists' newsletter.

The documents came from many different, proprietary or hard to find sources. The subscribers were willing to pay for the assured coverage, the selection, systematization and classification of the material, as well as the convenient presentation. Besides the magazine there came a database which started as a spin-off, containing documents from all issues of the newsletter, classified into various rubrics, which became an authoritative archive. This database is available as a CD-ROM with both full-text and keyword search. Recently, the archive can also be accessed via the Web.

Over the years, the category tree has evolved and grown organically. At some point, someone had the idea of designing a new, more consistent and more helpful category tree, but before it was finished someone else came up with a new set of keywords. Then came the realization that there was no way to retrospectively change the classification and re-classify all documents.

The use of automatic Text Categorization (aka Document Classification) in real-life applications is not just a matter of applying the best available technology.

To begin with, the nature of classification and the application opportunities it offers is not immediately obvious to potential customers. Some believe classification has to do with the recognition of certain keywords in the document, and confuse it with keyword extraction. Others believe that classification involves a deep understanding of the documents and expect too much from it. Modern categorization systems are somewhat in the middle, accurate but not clairvoyant. They have to learn each category from example documents.

Secondly, a categorization system is not of immediate and obvious benefit to millions of computer users. It must be packaged unobtrusively as a component of an application which is well-comprehensible to its intended users.

*Computing Science Institute, University of Nijmegen, The Netherlands, kees@cs.kun.nl

†Edmond Research & Development BV, Nijmegen, The Netherlands, Paul.Jones@edmond.nl

‡Fiscaal up to Date, Postbus 11620, 2502 AP Den Haag, The Netherlands.

And, lastly, a categorization system can not be introduced without hard effort. It must be embedded in a system which, besides its normal operational use, provides support for the transition from manual to automatic classification.

In this note we first introduce our own classification system and then discuss some of the problems involved in this transition. We argue the need for a careful bootstrap process, and show the results of a first comparison in user satisfaction between manual and automatic classification.

2 About the LCS system

The Linguistic Classification System LCS started its life in the Esprit project DORO as a research tool for

- improving classification algorithms (term selection, thresholding, multi-classification, hierarchical classification), and
- improving document representations by linguistic techniques (Head/Modifier frames).

In the European project PEKING¹, the LCS has been developed into an industry-quality automatic classification system intended to replace or support manual classification.

The LCS implements the Rocchio and Winnow algorithms, and uses uncertainty-based techniques for Term Selection [4]. Its datastructures have been optimized for large-scale applications, such as the classification of Patent Applications [3].

Its performance can be determined from the following measurements, made on two testsets:

- EPO1F: 16000 patent applications in 16 categories, totalling 461 Mbytes
- EPO1A: 16000 abstracts in 16 categories, totalling 21 Mbytes.

Each experiment consisted in training a 16-way classifier on 16000 documents and then testing it on the same documents. We give for two corpora the user-time on a 700Mhz Pentium IV under Linux and the amount of memory needed.

EPO1A		
Winnow	83.93	108.9MB
Rocchio	46.02	116.7MB
EPO1F		
Winnow	278.92 seconds	329.4MB
Rocchio	172.74	341.5MB

In applying it to business situations, we were soon faced with the bootstrap problem:

how to bootstrap from an existing manual classification to the intended (semi)automatic classification

¹<http://www.cs.kun.nl/peking>

2.1 Uncertainty

Given a reasonably large collection of documents pre-labeled with categories, many techniques are available for automatically learning a classifier that faithfully approximates the given labelling on the given documents and extends it to new documents [2].

The quality achieved is limited not so much by the technical properties of the learning algorithms as by the *uncertainty* in the manual labelling of the documents:

- judgements have to be yes/no, whereas the experts labeling the documents may discern more gradations
- the intended categories may not be so well distinguishable, so that the choice between categories are to a certain extent arbitrary
- documents may be classified on the basis of information known to the expert which is not represented in the document
- the nature of the different categories may vary from one expert to another, and it may vary over time [1]
- human experts make errors and show irreproducible behaviour.

The first cause of uncertainty is absent in the case of mono-classification (each document belongs to precisely one class), which may explain why in our experience mono-classification is much more accurate than multi-classification.

For these reasons, it is naive to believe that the bootstrap from the manual classification to automatic classification is just a matter of applying the training algorithm to (a subset of) the existing labeled documents labelled according to the existing category tree. Even with the best categorization system, in terms of Precision and Recall the result may then be disappointing.

3 The bootstrap process

A non-naive bootstrap process as we see it consists of four stages:

1. evaluation of the original classification, signaling documents that are unsuitable as training documents and categories that can not be distinguished well enough.

This evaluation has two purposes:

- for the user: to understand the current situation and to be able to improve the existing category tree
- for the system: to be able to clean up the train set by excluding documents which are not good examples for their category from the subsequent training process.

After a first training on the train set, two classes of unsuitable train documents can be distinguished:

- (a) unclassifiabiles
Documents obtaining very low scores for all categories, including the one(s) they are labeled with. Usually, such a document contains very few terms. Often it is just a short note, or a pointer to another document. Or it may wholly or partly be written in another language.
 - (b) outliers
Documents which obtain a higher score for another category than the one(s) they are labeled with.
2. elimination or re-labeling of unsuitable train documents.
Outliers may be eliminated from the train set, or re-labeled manually. Unclassifiable documents may be turned into *links* which belong to a category without being exemplary for it.
According to a threshold scheme, it is possible to treat the more obvious cases automatically while investing a measured amount of manpower in correcting errors and thus improving the resulting classification.
 3. possible revision of the category tree.
Any time invested in reconsidering and possibly revising the classification scheme is well spent. In fact, this is a normal (and probably overdue) act of maintenance. Therefore there should be support not only for the evaluation of the classification, but also for the revision of the category tree. It must be possible to split or join categories (together with their associated example documents) and to add, delete, shift or rename categories (with their example documents).
Various diagnostic aids (confusion matrix, distance measure, diagram of classifier performance over time) may be helpful.
 4. automatic induction of new classifier.
After the clean-up of the document collection and the category tree described above it is easy to train a new classifier automatically.
It is not necessary to train on all available documents, but the distribution of training examples over the categories should be representative for the documents expected during production.

4 An evaluation example

In the course of the PEKING project, two experiments will be performed in order to evaluate the applicability of automatic document classification techniques to the pre-classification of documents in the context of Fiscaal.

The intention of the experiments is not only to measure the objective technical properties of the LCS system (Precision and Recall on relevant documents from the User Fiscaal), but to help a (hypothetical) manager in choosing between two options:

1. to install a pre-classification system now, using the LCS classification technology that resulted from the DORO project and trained with the current category-tree on the present manually classified documents, or

2. first to perform a bootstrap and then use the classification technology available at the end of the PEKING project.

The technical improvements in the course of the PEKING project can be measured objectively, including the effect of the linguistic enhancements, but the important question (“will we be happy with the system”) needs a subjective evaluation, involving human users. It is not just a matter of Precision and Recall.

The question to investigate is not to what degree the automatic classification can mimic or approximate the current manual classification, but whether an automatic classification using the already classified documents as train documents is good enough for practical use, i.e. whether it is as least as useful to the intended users as the manual one. This can not be answered by an automatic evaluation but needs the involvement of experienced users to provide judgements on the category and index-terms assigned to a test set of documents which are sufficiently representative for the collection as a whole.

The usefulness of the manually and automatically assigned categories will be compared in a controlled experiment, in which the users are not informed about the source of the classification (blind experiment).

The evaluation has four phases:

1. selection of initial categories and documents
2. initial experiment to establish the baseline
3. selection of final categories and documents
4. final experiment to evaluate the usefulness of the PEKING system.

Of these, the first two have been performed.

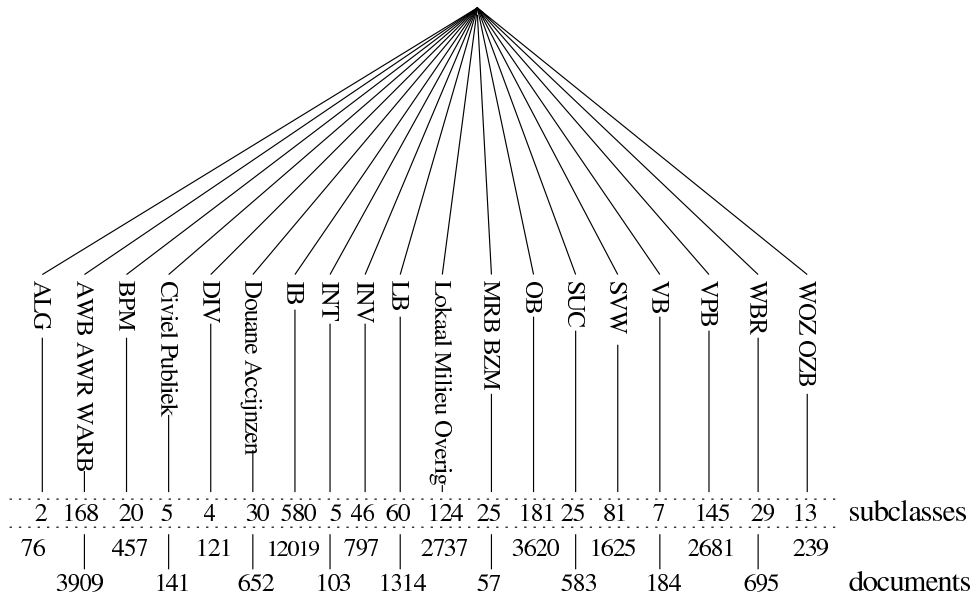
4.1 The corpus

The company Fiscaal (which takes part as a User in the Peking project) offers access to documents pertaining to fiscal law which are in principle in the public domain (tax laws and their explanatory documents, court decisions, decisions of the tax inspector) to which it adds value by systematic collection, classification and uniform presentation.

The classification has been performed manually for a number of years, based on two different approaches:

- a first classification into 19 tax categories
- assignation of keywords from a set of 5000.

The first classification is not quite a mono-classification, because some documents belong in multiple categories. The assignation of keywords is in fact a multiclassification, but the set of applicable keywords depends on the tax category, with many overlaps between categories. In the view of the user, the resulting classification is hierarchical rather than the simple cross-product of two orthogonal classifications, because the meaning of some keywords depend on the tax category.



The User has made available a corpus of 31000 docs (810 Mbytes) distributed unevenly over 19 main categories and more than 1500 subcategories (keywords), in a multiclassification with avg 3.4 categories per document. Each (sub)category had at least 20 example documents. These documents were split randomly into a train set (70%) and a test set (30% = 9518 documents). The LCS was trained on the selected train set and the test documents classified.

4.2 The objective test

In the objective part of the test, we measured Precision and Recall with the train- and testset described above.

For the first level multi-classification into 19 tax classes the results were as follows:

<i>class</i>	<i>precision</i>	<i>recall</i>
ALG	0.600	0.300
AWB AWR WARB	0.567	0.602
BPM	0.632	0.720
Civiel Publiek	0.422	0.563
DIV	0.480	0.632
Douane Accijnzen	0.729	0.729
IB	0.807	0.805
INT	0.000	0.000
INV	0.664	0.677
LB	0.579	0.583
Lokaal Milieu Overig	0.802	0.810
MRB BZM	0.761	0.806
OB	0.864	0.893
SUC	0.717	0.856
SVW	0.781	0.845
VB	0.521	0.578
VPB	0.694	0.765
WBR	0.815	0.826
WOZ OZB	0.426	0.650
MACRO average	0.642	0.665
MICRO average	0.767	0.739

One level lower, the micro-averaged Precision and Recall within each tax class were as follows:

<i>class</i>	<i>precision</i>	<i>recall</i>
ALG	0.700	0.368
AWB AWR WARB	0.425	0.303
BPM	0.515	0.345
Civiel Publiek	0.281	0.333
DIV	0.362	0.438
Douane Accijnzen	0.601	0.444
IB	0.480	0.353
INV	0.556	0.326
LB	0.477	0.344
Lokaal Milieu Overig	0.513	0.405
MRB BZM	0.618	0.356
OB	0.523	0.359
SUC	0.625	0.432
SVW	0.544	0.411
VB	0.432	0.500
VPB	0.537	0.383
WBR	0.691	0.539
WOZ OZB	0.429	0.288

This looks somewhat less convincing. The following table gives a resume of the accuracy of the entire hierarchy:

Number of documents in testset	9518
Number of documents automatically classified exactly as in pre-classification	685
Number of classifications in pre-classification	31985
Number of correct classifications	11565
Number of lost pre-classifications	20420
Number of found classes	11628
Precision	0.499
Recall	0.362

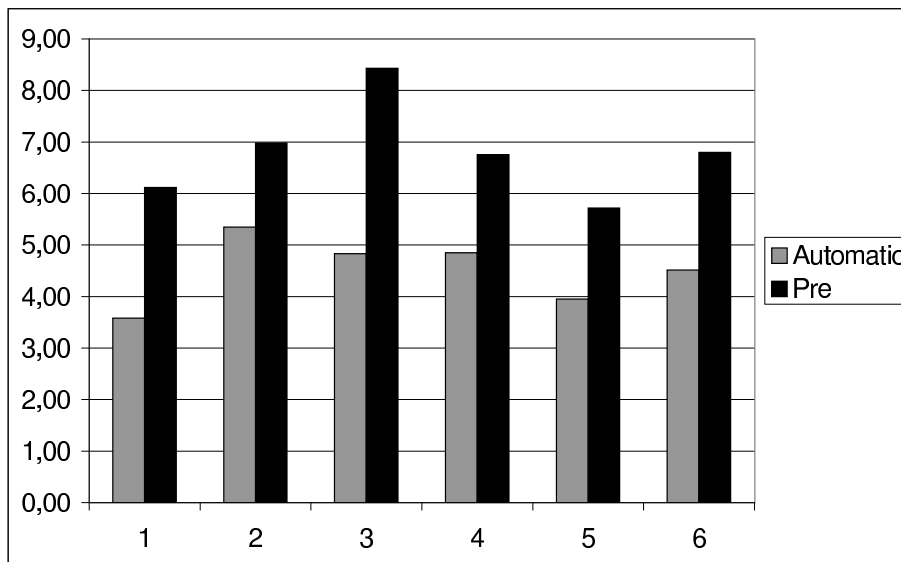
Notice that there were 31985 combinations <category, document> in the pre-classification but only 23694 in the automatic classification, because the LCS would rather give no classification than an uncertain one. These results were later somewhat improved by the use of shrinkage, but there is obviously a large difference between manual and automatic classification. Knowing that the original (manual) classification is less than perfect, this raises the question: how important is this difference to the user?

4.3 The subjective test

Making a random selection from the testset, a sequence of documents, together with their manual and automatic classifications, were offered to 6 human experts to answer the question whether one classification *is more useful* than the other or whether they are equally useful.

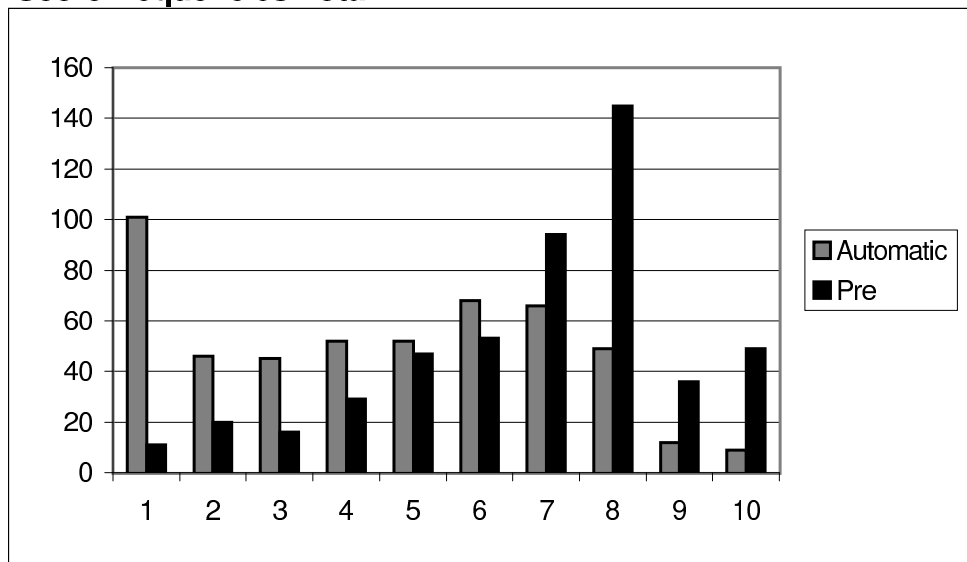
In case the manual and automatic classification of a test document fully agree, which will be the case for a small percentage of the documents, the document is not shown to the judges and it is counted as "equally useful".

In this blind experiment, the six experts mostly preferred the manual pre-classifications to the automatic ones:



The distribution of the notes given by the experts (1 is completely wrong, 6 = acceptable, 10 is excellent) is as follows:

Score frequencies Total



The automatic classification was not in every case inferior to the manual pre-classification:

$pre > aut$ 64%
 $pre = aut$ 18%
 $pre < aut$ 18%

It is obvious that the automatic classification, trained on the existing corpus without any improvement, is considered by all six experts to be less useful than the manual one. The average score given for the manual classification was 6.8, for the automatic one 4.5 - not a passing grade.

An important effect of this experiment was that it made clear to the User that the manual classification is also less than perfect. It is not the gold standard that has to be approximated as closely as possible.

4.4 Follow-up

In the bootstrap process after the initial experiment, the existing category tree will be analysed and restructured, and then used to bootstrap from the manual classification of all documents to the automatic classification of most documents. In this process categories may be renamed, split and/or joined, outlying documents reclassified and unclassifiable documents replaced by links. This makes it hard to compare the old manual classification of documents with their new automatic one. The set of final test documents will therefore have to be a fresh selection of documents, such that their former manual classification can be mapped to their new automatic one.

The final experiment after the bootstrap will be performed in a way similar to the initial experiment. It remains to be seen in how far a careful bootstrap and (by that time) improved classification techniques can improve upon the above objective and subjective results.

5 Conclusions

We have argued that technical excellence is not enough to make a Text Categorization system a useful component in document processing applications. The TC system must also provide support for the bootstrap from manual to automatic classification.

This bootstrap process can best be understood as *delayed maintenance*. The existing category tree with its representative documents must be analysed and revised, in order to make the tree more useful and consistent. This process has to be repeated as normal maintenance at regular intervals, albeit in a milder form: in comparison to manual classification, the work of an automatic classification system is much more transparent, but the documents to be assigned to a given category may change over time (“topic drift”) and changes in the environment may require modifications to the category tree.

We have described an initial experiment, which showed that in the case of Fiscaal a naive bootstrap straight from the existing pre-classified documents, without selection of train documents and reconsideration of the category tree, gave disappointing results, both from an objective and from a subjective point of view. As an analysis of the weaknesses of the current situation, this experiment proved very helpful. By means of a more careful bootstrap we hope to improve the quality of the classification so that the (semi)automatic system becomes at least as useful as the present costly and inflexible manual classification.

References

- [1] Avi Arampatzis, Jean Beney, C.H.A. Koster, Th.P. van der Weide, KUN on the TREC-9 Filtering Track: Incrementality, Decay, and Threshold Optimization for Adaptive Filtering Systems. The Ninth Text REtrieval Conference (TREC-9), Gaithersburg, Maryland, November 13-16, 2000.
- [2] Fabrizio Sebastiani, Machine learning in automated text categorization, ACM Computing Surveys, Forthcoming, 2002
<http://faure.iei.pi.cnr.it/~fabrizio/Publications/ACMCS02.pdf>
- [3] C.H.A. Koster, M. Seutter and J. Beney (2001), Classifying Patent Applications with Winnow, Proceedings Benelearn 2001, Antwerpen, 8pp.
- [4] C. Peters and C.H.A. Koster (2002), Uncertainty-based Noise Reduction and Term selection in Text Categorization, Proceedings ECIR 2002, Springer LNCS 2291, pp 248-267.