# Rigorous dimensionality reduction through linguistically motivated feature selection for text categorization

**Hans Friedrich Witschel** and **Chris Biemann**
NLP department
University of Leipzig
Augustusplatz 10–11
DE-04109 Leipzig
{witschel,biem}@informatik.uni-leipzig.de

## Abstract

This paper introduces a new linguistically motivated feature selection technique for text categorization based on morphological analysis. It will be shown that compound parts that are constituents of many (different) noun compounds throughout a text are good and general indicators of this text's content; they are more general in meaning than the compounds they are part of, but nevertheless have good domain-specificity so that they distinguish between categories. Experiments with categorizing German newspaper texts show that this feature selection technique is superior to other popular ones, especially when dimensionality is reduced substantially. Additionally, a new compound splitting method based on compact patricia tries is introduced.

## 1 Introduction

The task of automatic text categorization can be divided into two fields of research: first, appropriate features have to be selected for representing documents. Second, the actual classification algorithms have to be developed and applied to the previously generated feature vectors. Most of recent research has been devoted to the latter task.

In this paper, however, we argue that in text categorization, feature selection is absolutely crucial for the quality of classification results. Moreover, many applications require a drastic reduction in dimensionality, i.e. it is rarely possible or desirable to use the full set of terms occurring in a given document. Moreover, differences between feature selection algorithms become more visible as dimensionality is reduced (which is somewhat trivial because, when using the full set of available features from a text, all algorithms will have equal performance). We therefore consider feature selection for text categorization a good evaluation method for indexing algorithms that aim at very compact document descriptions.

As indicated by (Sebastiani, 2002), there are two possibilities of dimensionality reduction: selecting a subset of the existing terms or generating a set of synthetic terms, e.g. by using clustering or Latent Semantic Indexing (LSI). In this paper, an instance of term selection will be discussed.

It should be noted however, that the notion of ”term” or ”feature candidate” can be understood in various ways: in a bag-of-words model every string surrounded by whitespace will be considered a term – with the possible exception of so-called stop words. Alternatives are possible: as we will propose, compound constituents can also form a feature candidate set as well as phrases (multi-word units) or arbitrary character n-grams. Each method for generating a set of feature candidates can be individually combined with different selection methods for reducing its size.

In the following, we wish to make two major contributions:

- First, we introduce a new algorithm for feature selection that is based on shallow linguistic knowledge and especially designed to rigorously reduce dimensionality.
- Second, we support the findings of (Yang and Pedersen, 1997) who have shown that different algorithms for feature selection behave quite differently when the number of features is reduced significantly.

S. Werner (ed.), *Proceedings of the 15th NODALIDA conference, Joensuu 2005*, Ling@JoY 1, 2006, pp. 210–217
ISBN 952-458-771-8, ISSN 1796-1114. http://ling.joensuu.fi/lingjoy/
© by the authors

The rest of this paper is organized as follows: The following section introduces some related work, in section 3 the actual feature selection techniques that we want to compare will be discussed. Section 4 will detail one of the linguistic processing techniques that we used (namely compound splitting), sections 5 and 6 will describe the experiments that we conducted and section 7 concludes.

## 2 Related work

### 2.1 Statistical feature selection

Most approaches to feature selection rely on pure statistics. Normally a bag of words approach for representing documents is used together with these methods, i.e. all words (i.e. one-word units) from a text are used as feature candidates, regardless of their syntactic function (part-of-speech) or other linguistic characteristics. The only "linguistic" operation that is widely performed is the removal of so-called stop words (functional words) by predefined lists.

One of the simplest of these methods is selecting terms with medium to high *document frequency* (DF), i.e. ones that occur in many documents. However, terms with *very high* DF are normally excluded as stop words. A vocabulary that consists of terms with a medium to high DF is likely to cover a large portion of the collection, i.e. it is probable that each document contains at least one term from this vocabulary even if its size is reduced substantially. DF scores are used by e.g. (Ittner et al., 1995) or (Yang and Pedersen, 1997).

Some more sophisticated statistics are based on information-theoretic measures that select terms, the distribution of which is strongly biased towards documents from one single category (i.e. terms that occur in documents of one category but *not* in others). Examples for these measures include the $\chi^2$ measure (cf. e.g. (Yang and Pedersen, 1997; Galavotti et al., 2000)), information gain (Lewis, 1992; Larkey, 1998) or mutual information (Dumais et al., 1998; Larkey and Croft, 1996). This is only a very small fraction of all the research that has been carried out in that direction.

In a comparative study that evaluated many of the most popular statistical approaches, (Yang and Pedersen, 1997) surprisingly found DF to fall only very slightly short of the other, more sophisticated methods. Mutual information even performed significantly worse than DF. This means that the benefits of information-theoretic measures for feature selection in text categorization are somewhat arguable.

### 2.2 Linguistic methods

Linguistic methods for generating feature candidates have been applied in the past, but most efforts in this direction have concentrated on phrasal features: often noun phrases (identified in different ways – statistically or linguistically) are used as feature candidates (cf. e.g. (Lewis, 1992; Tzeras and Hartmann, 1993)). Different phrasal indexing approaches have led to different results, but most research in that direction found that the use of (noun) phrases as features does *not* improve classification accuracy because

> "an indexing language consisting of syntactic indexing phrases will have more terms, more synonymous or nearly synonymous terms, lower consistency of assignment (since synonymous terms are not assigned to the same documents), and lower document frequency for terms" (Lewis, 1992).

This has led to the general conclusion that linguistic feature selection methods should not be further explored.

Approaches that try to use linguistic information – apart from the identification of noun phrases – have therefore not attracted much attention. An example of such an approach can be found, however, in (Junker and Hoch, 1997), where the use of part-of-speech and additional morphological term characteristics is proposed: both of them were found to improve classification results on OCR and non-OCR texts.

As far as part-of-speech information is concerned, only nouns, adjectives and verbs were admitted as features in their experiments and morphological analysis comprised stemming and compound analysis. Parts of compounds were permitted as additional feature candidates (similarly to our hybrid strategy, see below) and mutual information was then applied as a statistical term selection method on this candidate set. (Junker and Hoch, 1997)

also found character n-grams to be good features (namely 5-grams), showing approximately equal performance to the use of the linguistic methods mentioned above.

The overall feature selection process in (Junker and Hoch, 1997) was similar to the one we are going to present in this paper, with the important difference that we are going to combine morphological analysis with a local statistical filter – instead of using the (global) mutual information measure – and use compound parts as the *only* feature candidates for describing texts.

## 3 Linguistically motivated feature selection

### 3.1 Preliminary thoughts

What should good features for text categorization look like? First, they should be specific of their domain or category – words or units that appear uniformly in texts throughout all categories are very ill suited for distinguishing between categories. This is the idea behind many of the statistical approaches introduced in the last section: measures like mutual information or $\chi^2$-tests aim at extracting "category-specific" features.

On the other hand, the selected vocabulary must cover as many documents as possible, i.e. each document should contain at least one term from the vocabulary. When reducing dimensionality through term selection techniques, however, documents must be described by only very few terms. This poses a serious problem: if terms are very specific, they are unlikely to cover a large portion of the document collection. Selecting terms with high document frequency has been proposed exactly for this reason: when reducing the size of the vocabulary significantly, the terms that we leave over must be general enough to cover the majority of all documents. This is also why weighting terms by TF/IDF is probably a bad idea: it prefers terms with high IDF, i.e. ones that occur in very few documents.

To summarize: good features for text categorization should be category-*specific*, but *general within* that category or domain.

The use of linguistic – or more precisely, shallow syntactic and morphologic – criteria that we propose is based on the intuition that some syntactic categories have a larger fraction of content-bearing elements than others. We especially focus on nouns and noun compounds because they tend to be more content-bearing and less ambiguous than verbs or adjectives.

More specifically, the parts of a compound noun (especially its head) have a more general meaning than the whole compound: "Saft" (juice) is more general than "Orangensaft" (orange juice). Therefore, compound constituents that appear frequently in many (different) compounds of a text tend to be good indicators of the text's *general topic*. Moreover, parts extracted from noun compounds are nearly always free morphemes or even words, i.e. they can appear in a text by themselves. They are thus also informative index terms when inspected by humans.

The approach that we will describe subsequently does not examine the distribution of compound parts throughout categories, i.e. it will not assure that they appear in feature vectors of only one category. Instead, a local feature selection technique using within-category frequencies will be used. We will see in the experiments that this is sufficient because compound parts are not only general but also specific of the topic that the documents cover: they yield surprisingly good classification results, especially at very low dimensionalities.

### 3.2 Feature Selection using compound constituents

The approach that we propose is based on syntactical as well as morphological knowledge: in a first step, common nouns are extracted by using a part-of-speech (POS) tagger and their frequencies are calculated. Thereafter, all these nouns are passed to a tool designed to split compounds into their constituents (see section 4). Whenever this tool produces two or more parts, i.e. whenever we find a true compound, a count for each of these parts is incremented by the frequency of the compound that contains it.

When regarding compound constituents and their counts as feature vectors, we can reduce dimensionality as follows: The whole set of positive training instances for each category is treated as one large document and compound parts are extracted from this text as indicated above. Then, we can select the X most frequent compound parts from each category as an indexing vocabulary. The feature vector for a sin-

gle text is computed by generating the list of all compound parts contained in both the compounds of this text and in the indexing vocabulary.

Splitting compounds is obviously restricted to languages which use one-word compounding, such as German, Dutch, Japanese, Korean and all Nordic languages. However, the same idea can in principle be applied to English as well: again using a part-of-speech (POS) tagger, it is possible to extract noun phrases that match POS patterns like N N (two successive nouns, e.g. "information retrieval") from texts. These often correspond to compounds in one-word compounding languages and their constituents can be treated in the same way as suggested for compound parts above.

## 4 Compound splitting

For setting up a compound splitting component, it is clearly desirable to use a machine learning approach: We would like to train a classifier using a set of training examples. In application, this classifier uses regularities acquired in the training phase to split compounds that have not been necessarily contained in the training set.

Generally, there are two ways to design a generic compound splitter: one is based on training on all possible breakpoints and using letter n-grams to both sides as features, e.g. used by (Yoon, 2000). Another way is to memorize possible prefixes and suffixes of compounds and match them during classification, a methodology conducted by e.g. (Sjöbergh and Kann, 2004). While n-gram splitters are capable of reaching comparatively high accuracy scores with small training sets, affix splitters need more training data but handle exceptions more naturally.

Here, we present an affix compound splitter that uses Compact Patricia Tries (CPT) as a data structure, which can be extended to function as a classifier on affixes of words.

### 4.1 Classification with Compact Patricia Tries

A trie is a tree data structure for storing strings, in which there is one node for every common prefix. The number of possible children is limited by the number of characters in the strings. Patricia tries (first mentioned in (Morrison, 1968)) reduce the number of nodes by merging all nodes having only one child with their parent's node. When using the structure for a string-based classification task, redundant subtrees and strings in leaves longer than 1 can be pruned, resulting in a structure called Compact Patricia Trie (CPT).

For classification, the sum of the weights for all classes in a subtree is stored with the string in the respective node. For example, in the CPT depicted in figure 1c), the prefix "Ma" has the class "m" associated with it three times, whereas the class "f" was seen only once with this prefix. Confidence of a node for a class C can be calculated by dividing the weight of C by the sum of the weights of all classes. Figure 1 shows an example, for thorough discussion on CPTs see e.g. (Knuth, 1999).

The CPT data structure possesses some very useful properties:

- the upper bound for retrieving a class for a word is limited by the length of the word and independent of the number of words stored in the CPT. When using hashes for subtree selection and considering limits on word lengths, search time is O(1).
- the number of classes for the classification task is not limited.
- when there is only one class per word, CPTs reproduce the training set: when classifying the previously inserted words, no errors are made.
- words that were not inserted in the CPT nevertheless receive a morphologically motivated guess by assigning the default class of the last matched node (partial match)

CPTs as classifiers can be put somewhat in between rule-based and memory-based learners. For unknown words, the class is assigned by choosing the class with the highest confidence in the node returned by a search. Nevertheless, CPTs memorize exceptional cases in the training set and therefore provide an error case library within the data structure.

### 4.2 Compound splitting with CPTs

Germanic compound nouns can consist of an arbitrary number of nouns or other word classes. A segmentation algorithm must proceed recursively, splitting the noun into parts that are split again until no more splitting can be performed. Segmentation can be done from the front and

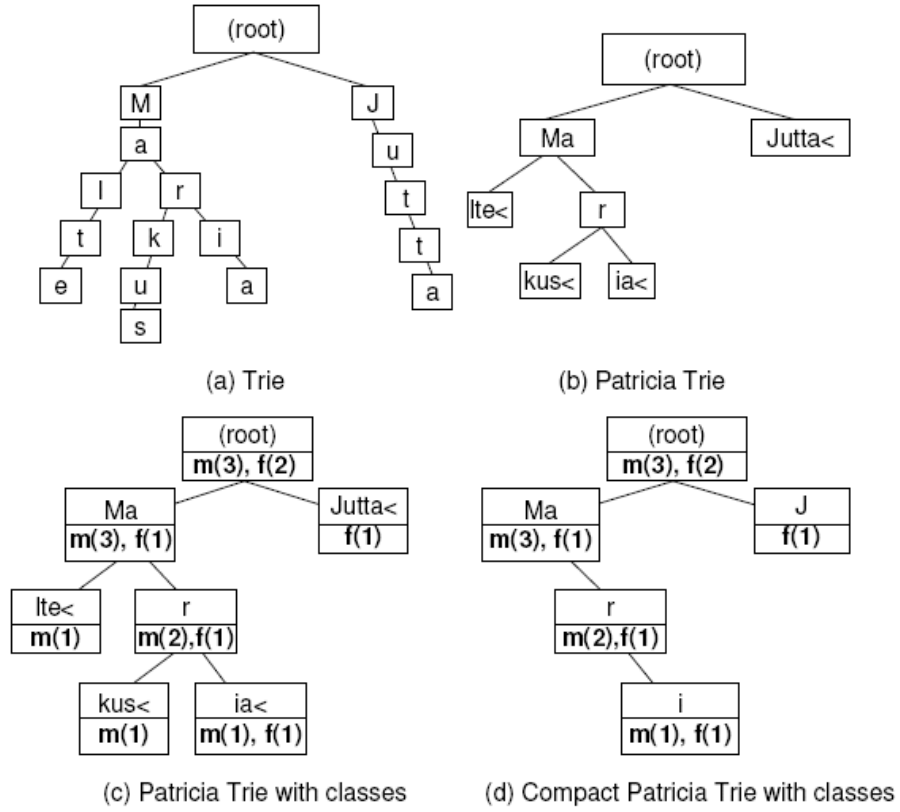


Figure 1: From Trie (a) to Patricia Trie (b,c) to CPT (d) for the classification of first name genders. m denotes male, f denotes female. Note that 'Maria' can be both.

from the end of the word. According to this, two CPTs are trained: One that memorizes at which position – counting from the beginning of the word – a split should be performed, and another one memorizing the break points in counting characters from the end of the word. The training set not only consists of known compound nouns, but also of all sub-compound nouns. Table 1 illustrates the training examples for both CPTs as obtained from the compound *Dampf//schiff//fahrt(s)//gesellschaft* (German lit: steam//ship//trip//society). The numbers indicate the position of the break points, the optional string part is used to denote possible interfixes (linking elements) that are inserted for phonological reasons, e.g. the (s) in table 1.

Now we have two classifiers predicting segmentation points on the basis of words. These classifiers either utter a proposal or respond "undecided" when confidence for the deepest retrieved node is too low. During segmentation, the following heuristics were applied:

- *Case 1*: both CPTs agree on a segmentation point - segment at this point
- *Case 2*: one of the CPTs is undecided - segment on the other's proposed point
- *Case 3*: the CPTs disagree: believe the CPT that reports the highest confidence
- *Case 4*: both CPTs are undecided or predict segmentation points out of word bounds: do not segment.

Evaluating the compound splitter using the Korean Compound noun training set of (Yun et al., 1997) with 10-fold cross-validation, we achieved an F-value of 96.32% on unseen examples and 99.95% on examples contained in the training set. The reasons for not perfectly reproducing the training set lies in the incapability of the approach to handle ambiguous splits (e.g. Swedish *bil+drulle* (bad driver) vs. *bild+rulle* (film roll)). These cases, however, do not play a major role in terms of frequency and can be handled by an exception list.

| **word** | **rule in prefix CPT** | **rule in suffix CPT** |
|---|---|---|
| dampfschifffahrtsgesellschaft | 5 | 12 |
| schifffahrtsgesellschaft | 6 | 12 |
| fahrtsgesellschaft | 5s | 12 |
| dampfschifffahrt | 5 | 5 |
| dampfschiff | 5 | 6 |
| schifffahrt | 6 | 5 |
| dampf | 5 | 5 |
| schiff | 6 | 6 |
| fahrt | 5 | 5 |
| gesellschaft | 12 | 12 |

Table 1: Compound constituents of *Dampfschifffahrtsgesellschaft*

For our experiments described in the next section, we used a German training set that was automatically constructed using a large corpus. Manual evaluation showed that more than 90% of segmentations are correct for compounds with at most 4 constituents.

## 5 Experimental setup

We conducted some experiments with a German newspaper corpus consisting of 3540 texts from 12 different subject areas using an implementation of Multinomial Naive Bayes from the Weka package[1] with 10-fold cross-validation. Experiments with other classifiers showed the same effects and are therefore omitted. We built three sorts of indexing vocabularies:

- *Compound parts:* For each category, the set of positive training instances was concatenated to form one single text and the parts occurring in many compounds throughout this text were extracted together with their frequencies.
- *Common nouns:* From the preliminary phase of our shallow linguistic analysis, we retained the set of common nouns, together with their frequencies. We used the same form of building the final feature set on these candidates, namely selecting the highest ranked nouns from each category.
- *DF:* A bag of words model without any linguistic knowledge, using document frequency (DF) for feature selection (which (Yang and Pedersen, 1997) have shown to behave well when compared to more sophisticated statistical measures, see section 2). Terms with medium to high DF were chosen in this method: the ones with *very high* DF (stop words) were first removed. Thereafter, terms with low DF were pruned in order to arrive at the different vocabulary sizes.

Finally, we implemented a *hybrid* strategy, combining nouns and compound parts, again selecting the most frequent items (nouns *or* compound parts) from each category.

## 6 Results

By varying thresholds, we produced results for different numbers of features. Figure 2 shows the classification accuracy for our three different feature selection techniques as a function of the indexing vocabulary size.

These results show that all algorithms perform similarly when using 1000 or more features (somewhat over 80% precision). When reducing the number of features drastically, however, the performance of the DF-based algorithm and the one with common nouns drops much faster than that of our compound part extraction.

When using as little as 24 features (i.e. only two from each category), DF term selection and common nouns both produce an accuracy of just around 35%, whereas when using compound parts, we obtain a precision of nearly 60%. This difference of performance can be understood when looking at the selected features: Table 2 shows the indexing vocabularies of size 24 for nouns and compound parts, detailing the

[1] http://www.cs.waikato.ac.nz/%7Eml/weka/

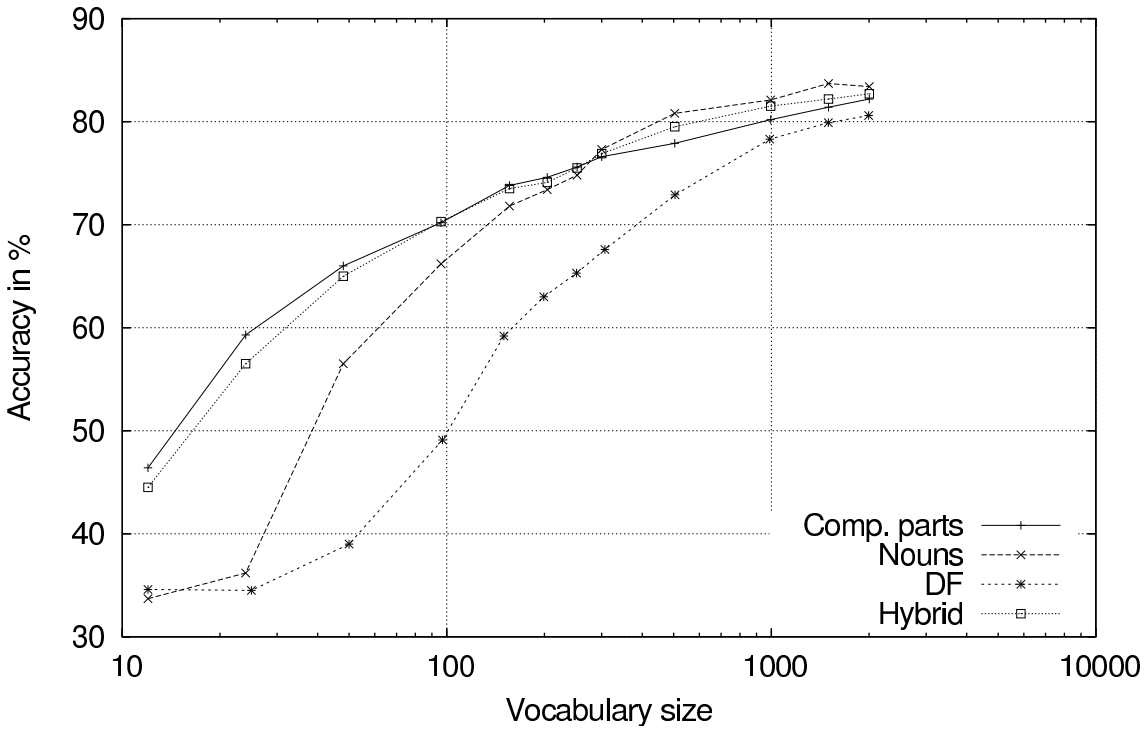


Figure 2: Classification results as a function of vocabulary size

features contributed by each of six (from 12) categories. Translations are – if necessary – given in brackets.

| **Category** | **Comp. parts** | **Nouns** |
|---|---|---|
| Auto (*cars*) | auto (*car*)<br>motor | Auto (*car*)<br>Jahr (*year*) |
| Erde (*earth*) | tier (*animal*)<br>meer (*sea*) | Jahr (*year*)<br>Tier (*animal*) |
| Geld (*money*) | gebühr (*fee*)<br>studie (*study*) | Student<br>Euro |
| Mensch (*man*) | zelle (*cell*)<br>stoff (*substance*) | Jahr (*year*)<br>Mensch (*man*) |
| Reise (*travel*) | stadt (*city*)<br>berg (*mountain*) | Jahr (*year*)<br>In (*in*) |
| Studium (*studies*) | schul (*school*)<br>uni (*university*) | Student<br>Jahr (*year*) |

Table 2: Top 2 features selected for six of the twelve categories by the *Nouns* and *Comp. parts* strategies

As we can see, the two most frequent nouns from the category "Job und Beruf" (job and profession) were "Jahr" (year) and "SPIEGEL" (the name of the magazine we built the corpus from). These occurred in many other categories as well. The two most prominent compound parts for the same category were "arbeit" (job) and "beruf" (profession) which is very specific to that domain (but, of course, also very general *within* that domain). This shows that compound parts are not only general but also domain-specific.

When using many features, however, the algorithm that uses common nouns performs slightly better than the one using compound parts. This suggests that the high generality of compound parts is at some point outperformed by the higher specificity of nouns. The hybrid strategy, combining compound parts and common nouns yielded good results but was still slightly inferior to using only nouns in the high dimensionality regions.

It would be interesting for future work to investigate if the statistical approaches like $\chi^2$-tests or information gain could further improve the results of the *Compound parts* strategy, e.g. in the higher (i.e. medium) dimension regions.

## 7 Conclusions

In this paper, some shallow linguistic techniques for feature selection were proposed and applied to text categorization. One of these

– namely the use of frequent compound constituents extracted from compound nouns – produces features of high "within-category" generality and acceptable domain-specificity.

All in all, we have been able to show two things: first, when reducing dimensionality substantially, there are notable differences between different feature selection algorithms. Second, we have built a selection algorithm that beats other approaches substantially when using a very low number of features.

This shows that although linguistic methods for feature selection have not been widely used in the past, it might be a good idea to so in the future, especially when dimensionality has to be reduced significantly.

## References

S.T. Dumais, J. Platt, D. Heckermann, and M. Sahami. 1998. Inductive learning algorithms and representations for text categorisation. In *Proceedings of CIKM-98*, pages 148–155.

Luigi Galavotti, Fabrizio Sebastiani, and Maria Simi. 2000. Experiments on the use of feature selection and negative evidence in automated text categorization. In *Proc. of ECDL '00*, pages 59–68.

D. J. Ittner, D. D. Lewis, and D. D. Ahn. 1995. Text categorization of low quality images. In *Symposium on Document Analysis and Information Retrieval*, pages 301–315.

Markus Junker and Rainer Hoch. 1997. Evaluating OCR and Non-OCR Text Representations for Learning Document Classifiers. In *ICDAR '97: Proceedings of the 4th International Conference on Document Analysis and Recognition*, pages 1060–1066.

Donald Knuth. 1999. *The Art of Computer Programming. Volume 3: Searching and Sorting*. Addison-Wesley, Reading, Massachusetts.

Leah S. Larkey and W. Bruce Croft. 1996. Combining classifiers in text categorization. In *Proc. of SIGIR '96*, pages 289–297.

Leah S. Larkey. 1998. Automatic essay grading using text categorization techniques. In *Proc. ofSIGIR '98*, pages 90–95.

David D. Lewis. 1992. An evaluation of phrasal and clustered representations on a text categorization task. In *Proc. of SIGIR '92*, pages 37–50.

D. Morrison. 1968. Patricia- practical algorithm to retrieve information coded in alphanumeric. *Journal of ACM*, 15(4):514–534.

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

Jonas Sjöbergh and Viggo Kann. 2004. Finding the correct interpretation of swedish compounds - a statistical approach. In *Proceedings of LREC-2004, Lisbon, Portugal*.

Kostas Tzeras and Stephan Hartmann. 1993. Automatic indexing based on bayesian inference networks. In *Proc. of SIGIR '93*, pages 22–35.

Yiming Yang and Jan O. Pedersen. 1997. A Comparative Study on Feature Selection in Text Categorization. In *ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning*, pages 412–420.

Juntae Yoon. 2000. Compound noun seqmentation based on lexical data extracted from corpus. In *Proceedings of the 6th Applied Natural Language Processing Conference*.

Bo-Hyun Yun, Min-Jeung Cho, and Hae-Chang Rim. 1997. Segmenting korean compound nouns using statistical information and a preference rule. *Journal of Korean Information Science Society (KISS)*, 24(8):900–909.