

Reducing False Positives by Expert Combination in Automatic Keyword Indexing

Anette Hulth

Department of Computer and Systems Sciences
Stockholm University
SE-164 40 Kista, Sweden
hulth@dsv.su.se

Abstract

This work extends previous work on automatic keyword indexing by showing how the number of incorrectly assigned terms—as measured by keywords assigned by professional indexers—may be highly reduced by combining the predictions of several classifiers. The classifiers were trained on different representations of the data, where the difference lay in the definition on what constitutes a potential keyword in a written document.

1 Introduction

The work described in this paper concerns automatic keyword indexing, where the goal is to automatically find a set of terms that describes the content of a document. The approach taken is that of supervised machine learning, i.e., a model (or a classifier) is constructed by training a learning algorithm on documents with known keywords. The model is subsequently applied to previously unseen documents, to select a suitable set of terms. This approach to automatically assign keywords is also used by, e.g., (Frank *et al.* 99; Turney 00; Pouliquen *et al.* 03). More specifically, this work concerns keyword *extraction*, i.e., the selected keywords are present in the document to which they are assigned. To evaluate the constructed models, manually assigned keywords are used as the gold standard.

Automatic keyword indexing is a difficult task, and the performance of the state-of-the-art keyword extraction is much lower than for many other NLP tasks, such as parsing and tagging. The low performance is partly due to the chosen evaluation method: It is to a certain extent inevitable that terms selected by a classifier differ from terms selected by a human indexer, as not even professional indexers agree on what set of terms best describes a document. One reason for this disagreement is that keyword indexing deals with natural language, and humans are often inconsistent in such tasks. This, in turn, is because

human languages allow for syntactic and semantic variations.

The work presented in this paper builds on work by (Hulth 03), in which several classifiers for automatic keyword extraction were evaluated. The most evident drawback with the classifiers was that they assigned too many terms that were not chosen as keywords by the human indexers. In some cases it may be desirable to have a model that assigns more terms than a human would do, for example if the classifier is to be used for semi-automatic indexing. In that case the goal for the training should be to find all manually assigned terms, as well as a limited set of additional terms. The final selection would then be made by a professional indexer. Semi-automatic indexing was, however, not the purpose of the experiments described in (Hulth 03).

In this paper, experiments on how the number of incorrectly assigned terms was reduced to a more acceptable level—as measured by keywords assigned by professional indexers—are described. The improvement was obtained by taking the majority vote of three classifiers which each was trained on a different representation of the data. The representations differed in how the terms were selected from the documents; using different definitions on what constitutes a potential keyword in a written text.

The outline of the paper is as follows: In the next section, a summary of the classifiers used for the expert combination presented in this paper is given. In Section 3, the ensemble technique and its results are described. Also, two approaches that did not work in reducing the amount of false positives¹ are shortly presented. Before concluding and giving some directions for future work, an example is given of the automatic keyword extraction before and after the expert combination.

¹A false positive is a term that has been given the label *positive* by the classifier although its true value is *negative*, i.e., it is not a manually assigned keyword.

2 Training the Classifiers

One of the most important aspects in machine learning is how the data are represented, and consequently what is given as input to the learning algorithm. In the case of keyword extraction, this basically means two things: How we define what constitutes a term in a written document, and what features of these terms that are believed to discriminate keywords from non-keywords.

In the experiments on automatic keyword extraction discussed in (Hulth 03), three different approaches to select the terms from the documents were used. These were all stemmed:

- uni-, bi-, and trigrams excluding stopwords (referred to as *n-grams*)
- noun phrase (NP) chunks
- words matching any of a set of part-of-speech (POS) tag sequences (referred to as *patterns*).

Three features were selected for the potential keywords. These were

- term frequency
- collection frequency (IDF)
- relative position of the first occurrence.

In addition, experiments with a fourth feature—that significantly improved the results—were performed for each term selection approach. This feature was

- the most frequent POS tag sequence assigned to the term.

In total, six models were evaluated on the test set (three term selection approaches with three and four features). The measures used for the evaluation were *precision*, *recall*, and *F-score* ($F_{\beta=1}$) for the selected keywords. To calculate the recall, the number of manually assigned terms actually present in the text was used, and a term was considered correct if its stemmed form was equivalent to a stemmed manually assigned keyword. The learning method applied was an implementation of *rule induction* using *divide-and-conquer* (RDS 03). The results from (Hulth 03) that are relevant for the experiments discussed in this paper are found in Table 1. These models were all trained on four features.

In the experiments described in this paper, the same data was used as in the previous experiments: A set of 2 000 abstracts from scientific journal papers with keywords assigned by professional indexers. Also, the division of the training (1 000 documents), validation (500 documents), and test (500 documents) sets was kept.

3 Combining the Experts

It has often been shown that combining experts leads to an improved accuracy, and there are several ways to apply ensemble techniques (see e.g., (Dietterich 98)). Basically, one may either manipulate the training data; for example in both *bagging* and *boosting* a number of classifiers are obtained by training on different subsets of the whole data set. Or, one may use different learning algorithms on the same training data to acquire different classifiers. There is also the question of how to combine the classifiers to consider, for example whether better performing classifiers should be given higher weights in the ensemble. (For a review on the latter, see for example (Bahler & Navarro 00).)

3.1 Combining Different Representations

In this section, an ensemble method that highly reduced the number of incorrectly assigned keywords, while still retaining a large number of correct terms, will be described. The ensemble was built from classifiers trained on different representations of the data. As mentioned in Section 2, six models were evaluated on the test set in (Hulth 03): Three term selection approaches with two sets of features (with or without the POS tag feature).

To reduce the number of incorrect terms a pairwise combination was initially made over these models, given that the term selection approaches were different. Thus, in total twelve pairs were obtained. In order to be considered a keyword, a term had to be selected by both models in the pair, i.e., an unanimity vote was used.

The twelve pairs were evaluated on the validation data, and the three pairs (one pair for each combination of the term selection approaches) with the highest precision were selected. The reason for choosing precision as the selection criteria was that the goal is to reduce the false positives, thus the proportion of these should be as small as possible.

Method	Assign. tot.	Assign. \bar{x}	Corr. tot.	Corr. \bar{x}	Prec.	Recall	F-score
<i>n</i> -grams	7 815	15.63	1 973	3.95	25.2	51.7	33.9
NP-chunks	4 788	9.58	1 421	2.84	29.7	37.2	33.0
Patterns	7 012	14.02	1 523	3.05	21.7	39.9	28.1

Table 1: For each term selection approach is shown: The number of assigned (Assign.) terms in total (tot.) and mean (\bar{x}) per document; the number of correct (Corr.) terms in total and mean per document; precision; recall; and F-score. The total number of manually assigned terms present in the abstracts in the test data is 3 816, and the mean is 7.63 terms per document.

The three pairs that were selected were

- *n*-grams with the PoS tag feature + NP-chunks with the PoS tag feature
- *n*-grams with the PoS tag feature + patterns with the PoS tag feature
- NP-chunks with the PoS tag feature + patterns with the PoS tag feature,

in other words, the three term selection approaches with the PoS tag feature. In Table 2, the results for these three combinations on the test set are shown: *n*-grams + NP-chunks assign the set of 500 abstracts in total 2 004 keywords, i.e., on average 4.01 terms per document. Of these are on average 1.80 terms correct. If looking at the actual number of keywords assigned, 27 documents have 0 terms, while the maximum number of terms assigned is 21. The median is 4.

For the *n*-grams + patterns pair, in total 3 822 keywords are assigned. Of the 7.64 terms on average per document, 2.14 are correct. If examining the actual distribution, 8 documents have no keywords assigned, the maximum is 34, and the median is 7.

Finally, the NP-chunks + patterns assign in total 1 684 keywords, i.e., on average 3.37 terms per document; of these are 1.42 correct. The maximum number of terms actually assigned is 14. 32 documents have 0 terms, and the median is 3.

As can be seen in this table, the precision has increased for all pairs, compared to the performance of the individual classifiers (see Table 1). However, the F-score has decreased for all three combinations. As it is important not only to assign correct terms, but also to actually find the manually assigned keywords, recall is considered equally important (the reason for when calculating the F-score giving β the value 1).

As these results were still not satisfactory from the point of view of the F-score, an experiment

with joining the results of the three pairs was performed, i.e., by taking the union of the terms assigned by the pairs. Doing this is in fact equivalent to taking the majority vote of the three individual classifiers in the first place. These results are shown in Table 3. In total, 5 352 terms are then assigned to the test set. On average per document, 3.31 terms are correct, and 7.39 are incorrect. The actual assignment of the terms varies between 41 and 0 for four documents, and the median is 10 keywords per document.

These values should be compared to the number of keywords assigned to the test set by the professional indexers. Three documents have no keywords present in the abstract. The median is 7 terms, and the maximum is 27. There are in total 3 816 manually assigned keywords. The F-score when taking the majority vote for the three classifiers is 36.1, thus higher than for any of the individual classifiers. The precision is also higher than for any of the component models, and the recall is higher than for two of the individual classifiers. If comparing this result to the *n*-gram approach, that has the highest F-score, the number of false positives has decreased by 2 145 terms, while 318 true positives are lost.

As another improvement, the subsumed terms may be removed, i.e., if a term is a substring of another assigned keyword, the substring is removed. In Table 3 one can see that although some correctly assigned terms are removed as well (5.9%), the number of false positives decreases by 24.0%. If looking at the actual distribution on the test set, four documents have 0 terms. The maximum number of terms assigned is 30, while the median is 8 keywords per document. This results in the highest F-score (38.1) obtained on the test set for these experiments.

Method pair	Assign. tot.	Assign. \bar{x}	Corr. tot.	Corr. \bar{x}	Prec.	Rec.	F-score
<i>n</i> -gram+Chunk	2 004	4.01	902	1.80	45.0	23.6	31.0
<i>n</i> -gram+Pattern	3 822	7.64	1 069	2.14	28.0	28.0	28.0
Chunk+Pattern	1 684	3.37	708	1.42	42.0	18.6	25.7

Table 2: The number of assigned (Assign.) terms in total (tot.) and mean (\bar{x}) per document; the number of correct (Corr.) terms in total and mean per document; precision; recall; and F-score for the three best performing pairs, evaluated on the test set. The total number of manually assigned terms present in the abstracts in the test data is 3 816, and the mean is 7.63 terms per document.

Majority vote	Assign. tot.	Assign. \bar{x}	Corr. tot.	Corr. \bar{x}	Prec.	Rec.	F-score
Sub. not removed	5 352	10.70	1 655	3.31	30.9	43.4	36.1
Sub. removed	4 369	8.74	1 558	3.12	35.7	40.8	38.1

Table 3: The number of assigned (Assign.) terms in total (tot.) and mean (\bar{x}) per document; the number of correct (Corr.) terms in total and mean per document; precision; recall; and F-score applying the majority vote, without and with subsumed terms (Sub.) removed, evaluated on the test set. The total number of manually assigned terms present in the abstracts in the test data is 3 816, and the mean is 7.63 terms per document.

3.2 Lessons Learned

Before any successful results were obtained on the task of reducing the number of incorrectly assigned keywords, two other methods were examined. In these two experiments, combinations of classifiers were made for each term selection approach separately. In the first experiment, *bagging* (Breiman 96) was applied, i.e., from a training set consisting of n examples², a new set of the same size is constructed by randomly drawing n examples with replacement. This procedure is repeated m times to create m classifiers. Both voting, with varying numbers of classifiers that should agree, as well as setting varying threshold values for the number of maximum terms to assign to each document in combination with voting, was tried.

In the second unsuccessful experiment, the fact that the data set is unbalanced was exploited. By varying the weights given to the positive examples for each run, a set of classifiers was obtained. Thereafter voting was applied in the same fashion as for the first unsuccessful experiment. In addition, a simple weighting scheme was applied, where higher weights were given to classifiers that found more correct terms.

As the results for these experiments were poor, they are not presented in this paper. Although the number of false positives did decrease, too

many of the true positives also disappeared. As these experiments did not succeed, it may be suspected that the selected features are not enough to discriminate keywords from non-keywords, at least not in this collection.

3.3 An Example of Automatically Assigned Terms

An example of the automatic extraction will now be given. The example is selected from the test set, and is one of four documents that have three, four, and seven keywords respectively assigned per combined pair; these values correspond to the median values for the three pair-wise combinations. In Figure 1, the abstract is shown together with both the manually assigned keywords, as well as with the automatically assigned terms using the majority vote as described in this paper, with the subsumed terms removed. In Figure 2, all potential keywords extracted by the three different terms selection approaches are displayed, before the classifiers are applied. In Figure 3, the keywords selected by each individual classifier for this abstract are shown. Finally in Figure 4, the keywords for the three pairs are displayed. In the three figures, the terms in bold are manually assigned keywords. As can be seen from the large number of terms in Figure 2, a machine learning component is crucial for restricting the number of terms assigned.

²An *example* is a feature value vector for, in this case, each potential keyword.

Abstract:
Lung metastasis detection and visualization on CT images: a knowledge-based method. A solution to the problem of lung metastasis detection on computed tomography (CT) scans of the thorax is presented. A knowledge-based top-down approach for image interpretation is used. The method is inspired by the manner in which a radiologist and radiotherapist interpret CT images before radiotherapy is planned. A two-dimensional followed by a three-dimensional analysis is performed. The algorithm first detects the thorax contour, the lungs and the ribs, which further help the detection of metastases. Thus, two types of tumors are detected: nodules and metastases located at the lung extremities. A method to visualize the anatomical structures segmented is also presented. The system was tested on 20 patients (988 total images) from the Oncology Department of La Chaux-de-Fonds Hospital and the results show that the method is reliable as a computer-aided diagnostic tool for clinical purpose in an oncology department.
Manually assigned keywords:
computed tomography; computer-aided diagnostic tool; ct images; image interpretation; knowledge-based top-down approach; lung metastasis detection; oncology; thorax; three-dimensional analysis
Automatically assigned keywords:
clinical purpose; computed tomography; computer-aided diagnostic tool; ct images; image interpretation; knowledge-based top-down; la chaux-de-fonds hospital; lung metastasis detection; oncology; radiotherapist; three-dimensional analysis; top-down approach; total images

Figure 1: An example of an abstract with keywords, both manually assigned keywords present in the abstract, and automatically assigned keywords, using the majority vote as described in this paper.

4 Conclusions and Future Work

The experiments and the evaluation presented in this paper concerns automatic keyword extraction. I have here shown how the number of automatically assigned keywords that are incorrect may be highly reduced, while the number of correct terms is still satisfactory, as measured by keywords assigned by professional indexers. The improvement is achieved by taking the majority vote for three classifiers, each trained on a different representation of the data. The representations differ in how the terms are selected from the documents. The three methods used are uni-, bi-, and trigrams; NP-chunks; and terms matching any of a set of POS patterns (see (Hulth 03) for details). If precision for some reason is considered more important than the F-score—if the assigned keywords must have a high quality—a combination of either n -grams and NP-chunks or NP-chunks and patterns using an unanimity vote should be used instead.

In order to establish which errors that are specific for one term selection approach, while not present in the two other (i.e., which types of errors are avoided by taking the majority vote), all

false positives from ten arbitrarily selected documents in the validation set were collected. Unfortunately, it was difficult to categorise the errors made by each approach (as may be suspected when looking at Figure 3). One of the few things that could be noted was that some of the terms selected by the NP-chunk classifier began with a determiner (e.g., ‘a given grammar’). These terms are rarely keywords, and are not extracted by the two other approaches (most determiners are stop-words, and are not part of the POS patterns). However, a more thorough investigation must be made, where also the selected terms for each classifier are compared to all potential keywords extracted from the documents before the classifier is applied, to first establish which terms that are filtered out already on the classification level.

An alternative to taking the majority vote as presented in this paper, could be to rank the terms according to how many of the individual classifiers that agree upon a term being a keyword, thus obtaining a probabilistic output. First the classifiers would need to be ranked internally according to their previous performance. A threshold value for the maximum number of key-

<i>n</i>-grams
algorithm; algorithm first detects; analysis; analysis is performed; anatomical; anatomical structures; anatomical structures segmented; approach; approach for image; chaux-de-fonds; clinical; clinical purpose; computed; computed tomography ; computer-aided; computer-aided diagnostic; computer-aided diagnostic tool ; contour; ct; ct images ; department; department of la; detection; detection and visualization; detection of metastases; detection on computed; detects the thorax; diagnostic; diagnostic tool; extremities; followed; help; help the detection; hospital; image interpretation ; images; images before radiotherapy; inspired; interpret ct; interpret ct images; interpretation; knowledge-based; knowledge-based method; knowledge-based top-down; knowledge-based top-down approach ; la; la chaux-de-fonds; la chaux-de-fonds hospital; located; lung; lung extremities; lung metastasis; lung metastasis detection ; manner; metastases; metastases located; metastasis; metastasis detection; method; method is inspired; method is reliable; method to visualize; nodules; nodules and metastases; oncology ; oncology department; patients; performed; planned; purpose; radiologist; radiologist and radiotherapist; radiotherapist; radiotherapist interpret; radiotherapist interpret ct; radiotherapy is planned; radiotherapy; reliable; results; ribs; scans; segmented; solution; structures; structures segmented; system; system was tested; tested; tested on patients; thorax ; thorax contour; three-dimensional; three-dimensional analysis ; tomography; tool; tool for clinical; top-down; top-down approach; total; total images; tumors; tumors are detected; two-dimensional; two-dimensional followed; types; types of tumors; visualization on ct; visualize; visualize the anatomical;
NP-chunks
20 patients; 988 total images; a computer-aided diagnostic tool; a knowledge-based method; a knowledge-based top-down approach; a method; a radiologist; a solution; a three-dimensional analysis; a two-dimensional; an oncology department; clinical purpose; computed tomography ; ct; ct images ; image interpretation ; la chaux-de-fonds hospital; lung metastasis detection ; metastases; nodules; radiotherapist; radiotherapy; the algorithm; the anatomical structures; the detection; the lung extremities; the lungs; the manner; the method; the oncology department; the problem; the results; the ribs; the system; the thorax; the thorax contour; tumors; two types; visualization; which;
Patterns
988 total; algorithm; analysis; anatomical; anatomical structures; approach; approach for image; chaux-de-fonds; chaux-de-fonds hospital; clinical; clinical purpose; computed; computed tomography ; computer-aided; computer-aided diagnostic; computer-aided diagnostic tool ; contour; ct; ct images ; ct scans; department; detection; diagnostic; diagnostic tool; extremities; hospital; image interpretation ; images; interpretation; knowledge-based; knowledge-based method; knowledge-based top-down; knowledge-based top-down approach ; la; la chaux-de-fonds; la chaux-de-fonds hospital; lung; lung extremities; lung metastasis; lung metastasis detection ; manner; metastases; metastasis; metastasis detection; method; nodules; oncology ; oncology department; patients; problem; problem of lung; purpose; radiologist; radiotherapist; radiotherapy; reliable; results; ribs; solution; structures; system; thorax ; thorax contour; three-dimensional; three-dimensional analysis ; tomography; tool; top-down; top-down approach; total; total images; tumors; two-dimensional; types; visualization;

Figure 2: Potential terms according to each term selection approach.

words to assign to each document would then have to be set, either by the system or by a user.

When inspecting the automatically assigned terms, it may be conclude that using keywords

assigned by one professional indexer as the gold standard for the evaluation does not always give justification to the models. Many automatically selected keywords have a clear relation to the sub-

<i>n</i>-grams
chaux-de-fonds; clinical purpose; computed tomography ; computer-aided diagnostic tool ; ct images ; knowledge-based method; knowledge-based top-down; knowledge-based top-down approach ; la chaux-de-fonds hospital; lung extremities; lung metastasis; metastases; metastasis detection; oncology ; oncology department; ribs; three-dimensional analysis ; top-down approach; total images
NP-chunks
988 total images; clinical purpose; computed tomography ; ct; ct images ; image interpretation ; la chaux-de-fonds hospital; lung metastasis detection ; radiotherapist
Patterns
chaux-de-fonds hospital; computer-aided diagnostic tool ; department; image interpretation ; knowledge-based top-down; la chaux-de-fonds; lung metastasis; lung metastasis detection ; oncology ; radiotherapist; thorax contour; three-dimensional analysis ; top-down approach; total images; tumors

Figure 3: Assigned terms for each term selection approach before the expert combination. (The terms in bold are also manually assigned terms.)

<i>n</i>-grams + NP-chunks
clinical purpose; computed tomography ; ct images ; la chaux-de-fonds hospital
<i>n</i>-grams + patterns
computer-aided diagnostic tool ; knowledge-based top down; lung metastasis; oncology ; three-dimensional analysis ; top-down approach; total images
NP-chunks + patterns
image interpretation ; lung metastasis detection ; radiotherapist

Figure 4: Assigned terms for each pair of the term selection approaches. (The terms in bold are also manually assigned terms.)

ject at hand, although not chosen by the human for one reason or another. Alternatives to this type of evaluation are currently under investigation.

Acknowledgements

For valuable comments and suggestions: Beáta Megyesi, and Tony Lindgren.

References

- (Bahler & Navarro 00) Dennis Bahler and Laura Navarro. Methods for combining heterogeneous sets of classifiers. In *17th National Conference on Artificial Intelligence (AAAI 2000): Workshop on New Research Problems for Machine Learning*, 2000.
- (Breiman 96) Leo Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.
- (Dietterich 98) Thomas G. Dietterich. Machine learning research: Four current directions. *The AI Magazine*, 18(4):97–136, 1998.
- (Frank *et al.* 99) Eibe Frank, Gordon W. Paynter, Ian H. Witten, Carl Gutwin, and Craig G. Nevill-Manning. Domain-specific keyphrase extraction. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'99)*, pages 668–673, Stockholm, Sweden, 1999.

(Hulth 03) Anette Hulth. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2003)*, pages 216–223, Sapporo, Japan, 2003.

(Pouliquen *et al.* 03) Bruno Pouliquen, Ralf Steinberger, and Camelia Ignat. Automatic annotation of multilingual text collections with a conceptual thesaurus. In *Proceedings of the Workshop on Ontologies and Information Extraction*, Bucharest, Romania, July 2003.

(RDS 03) RDS. Rule Discovery System, Compumine AB, 2003. www.compumine.com.

(Turney 00) Peter D. Turney. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336, 2000.