Also publ. in: Natural Language Processing and Information Systems : 14th International Conference on Applications of Natural Language to Information Systems, NLDB 2009, Saarbrücken, Germany, June 24-26. / ed. by Helmut Horacek ... Berlin : Springer, 2010. pp. 154-168
The original publication is available at www.springerlink.com

Analyzing document collections via context-aware term extraction

Daniel A. Keim, Daniela Oelke and Christian Rohrdantz

University of Konstanz, Germany
firstname.lastname@uni-konstanz.de

Abstract. In large collections of documents that are divided into predefined classes, the differences and similarities of those classes are of special interest. This paper presents an approach that is able to automatically extract terms from such document collections which describe what topics discriminate a single class from the others (discriminating terms) and which topics discriminate a subset of the classes against the remaining ones (overlap terms). The importance for real world applications and the effectiveness of our approach are demonstrated by two out of practice examples. In a first application our predefined classes correspond to different scientific conferences. By extracting terms from collections of papers published on these conferences, we determine automatically the topical differences and similarities of the conferences. In our second application task we extract terms out of a collection of product reviews which show what features reviewers commented on. We get these terms by discriminating the product review class against a suitable counter-balance class. Finally, our method is evaluated comparing it to alternative approaches.

1 Introduction

With the growing amount of textual data available in digital form, also methods and techniques for exploring these resources are increasingly attracting attention. In many cases classes (or clusters) of documents can be distinguished and topical differences and similarities among those classes are of interest. Depending on the concrete task it can also be worthwhile to explore stylistic or linguistic differences.

In this paper we present an approach that helps in analyzing a set of classes of documents with respect to the question what one class of documents discriminates from the rest - by extracting discriminating terms. The technique also determines so-called overlap terms that discriminate a subset of the classes from the remaining ones. The classes of documents e.g. could correspond to different scientific conferences with their published papers as documents. The extracted discriminating terms then show the topics that are unique for the specific conference. See figure 1 for an example that was generated by our new approach. In the venn diagram each circle represents one conference. All three conferences deal with graphical representations and visualization. Yet, each conference has its own specific orientation in the field. In an outer section of the diagram that is unique for one of the conferences the terms that discriminate this conference from all the others are displayed. In the case of the Vis and the comparison to Siggraph and InfoVis those terms are {flow field, scalar field, volume data, volume

dataset, vector field, volume visualization}. Furthermore, you can see the terms that are shared by two conferences and discriminate them against the third conference in the overlap regions of the diagram. Apparently, there is no overlap of the Siggraph and the InfoVis conference. While this might not be surprising for an expert in the area of these conferences (as the Vis conference is topically somewhere in between Siggraph and InfoVis), it provides quite useful information to non-experts without requiring substantial reading efforts. The overlap area of all three conferences remains empty, because our approach only extracts discriminating terms and in this case there is nothing to discriminate against. Please note that this is a very small introductory example. In section 4.1 we present the result of an analysis with more conferences and more terms.
But the application area of our method is much wider than that. The technique can be applied to any application task in which discriminating and / or overlapping terms between different classes are of interest. It turns out to be a powerful tool in any scenario in which terms that cover a certain topic or aspect have to be separated not only from general stopwords but also from terms covering other aspects that are currently not of interest. An example for such a scenario is given in section 4.2 where we use our technique to extract product attributes from a set of printer reviews. The challenge here is to extract the terms that hold the information about what the customers were satisfied or dissatisfied with, but filter out the review-typical words that they use to convey their message.

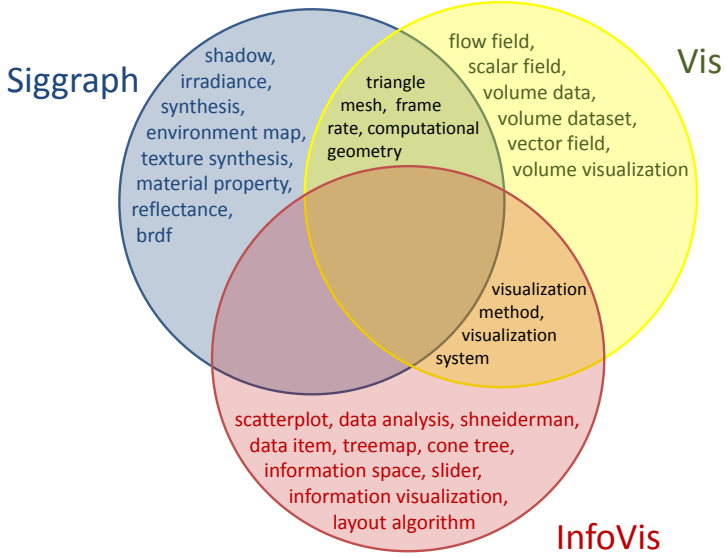

Fig. 1: Discriminating and overlapping terms for the three conferences Siggraph, Vis, and InfoVis (generated with about 100 papers of each conference). Terms in the overlapping areas are shared by two classes and discriminate them against the third one while the rest of the terms discriminates one specific class against the others.

2 Related Work

Numerous methods are dedicated to the extraction of terms out of document collections. These methods can be divided into four main categories: keyword extraction methods, information extraction methods, labeling methods and domain specific term extraction methods.

Approaches for keyword extraction often originate from the information retrieval field like e.g. the prominent TFIDF method ([1], [2]). An extensive survey on that can be found in [3]. But also in text mining research keyword extraction methods play a role ([4], [5]). In a usual case there is a measure that allows to score terms with respect to a document or a document collection and a certain number of top scored terms are then extracted.

An important example for information extraction is the named entity recognition. It is aimed at extracting proper names, that have a certain semantic category, in order to construct semantic lexica ([6], [7]). Typical examples for such categories are names of persons, companies or locations.

Among the term extraction approaches are some that extract domain specific terms comparing an analysis corpus of a certain domain with a reference corpus. The reference corpus is aimed to be as broad and universal as possible and can either be a general language corpus ([8], [9]) or composed of several other domain corpora [10]. Another approach takes a large collection of heterogeneous newspaper articles as a reference corpus [11]. Those approaches are useful for example to support terminology extraction or ontology construction.

Methods for labeling are mainly used for visualization tasks. Usually they extract very few terms that describe (the documents that constitute) a certain area of a visualization. In the ThemeScape^{TM} visualization [12] a common TFIDF approach is used for the labeling of the distinct document clusters. A similar labeling approach is done in the WEBSOM visualization [13] where the relative frequencies of terms in the different nodes of the self-organizing map are compared [14] [15].

Our method is similar to the approaches that do domain specific term extraction because we also compare the scores of a certain term for different domains/classes. Yet, in contrast to those methods we compare several class corpora with and to each other instead of using a general reference corpus for comparison. Furthermore, we use a novel measure called TFICF which is an adaption of the popular TFIDF measure to assess the importance of a term within a class. This allows us to determine discriminating terms for single classes or sets of classes in the concrete context of other interesting classes. By doing so, we are able to figure out the topical coherences and distinctions among a whole set of particular classes and thus satisfy a very specific information need.

In [16] a term frequency inverse cluster frequency value is calculated to get feature vectors of previously attained clusters of document paragraphs. In contrast to our approach the cluster simply can be seen as a concatenation of all of its documents so that actually there is no difference to the common TFIDF formula.

Our approach is also situated in the context of contrastive summarization [17] and comparative text mining [18] which is a subtask of contextual text mining [19]. Contrastive summarization has a rather narrow application field, it only regards the binary case (two classes) and is focused on opinion mining. Having reviews of two products, the aim is

to automatically generate summaries for each product, that highlight the difference in opinion between the two products.
While the fundamental idea of comparative text mining is closely related to our work, the outcome of the cross-collection mixture model proposed in [18] is rather orthogonal to our approach. The process is subdivided in two steps "(1) discovering the common themes along all collections; (2) for each discovered theme, characterize what is in common among all the collections and what is unique to each collection". Whereas this kind of analysis is based on the themes common to *all* classes, our method does explicitly not account for those themes, but for themes that discriminate one or several classes from the remaining ones.

The rest of this paper is organized as follows. First, in section 3 we motivate and introduce our new technique. Then in section 4 we present two application domains with concrete examples. In section 5 we provide an evaluation of our methods and compare it with other techniques. Finally, in section 6 we give a conclusion.

3 Technique

In order to determine discriminating or overlap terms, first of all we need to be able to quantify how important a certain term is for a certain class. It would be straight forward to use the standard TFIDF method. But unfortunately that approach is not suitable in this case. The TFIDF value determines an importance value for a certain term with respect to a document within a document collection. But what we need is an importance value for a certain term with respect to a whole document collection within the context of other document collections. Subsection 3.1 introduces the concept of TFICF (term frequency inverse class frequency) an extension of TFIDF that fulfills our criteria. In subsection 3.2, we explain how to use this new measure to extract discriminating and overlap terms. Finally, some notes on parameter tuning and preprocessing are given in the subsections 3.3 and 3.4.

3.1 Term frequency inverse class frequency (TFICF)

TFICF (term frequency inverse class frequency) is an extension of the classical TFIDF measure. The formula for weighted TFICF is composed of three factors (see equation 1).

$$\forall \text{ terms } t_i \wedge \forall \text{ classes } C_j \text{ with } i \in \{1 \cdots \#terms\} \text{ and } j \in \{1 \cdots \#classes\} : \text{[1]}$$

$$weighted - tficf(t_i, C_j) = distr\_weight(t_i, C_j) \cdot tf(t_i, C_j) \cdot icf(t_i) \tag{1}$$

The tf value reflects the normalized overall frequency of a term within a collection. The icf (inverse class frequency) value takes into account in how many classes the term is

[1] "#" stands for "number of all..."

present. The distr_weight value depends on the distribution of a term over the documents of a collection.
The tf value is calculated dividing the overall frequency of a term among the documents of a collection by the overall number of tokens in the collection (see equation 2).

$$tf(t_i, C_j) = \frac{\sum_{k=1}^{\#docs_j} freq(t_i, doc_{jk})}{\sum_{k=1}^{\#docs_j} \#tokens(doc_{jk})} \tag{2}$$

The rationale is that longer documents exert a stronger influence on the tf value than shorter documents, which is appropriate in most application scenarios. However, the influences of all documents may be adjusted to be similar by considering only the relative frequency of terms in documents.
In contrast to the standard idf formula our icf formula has to operate on classes of documents instead of single documents. A straight forward application of the idf formula would be to say that a term t is an element of a class C, if it occurs in at least one of the corresponding documents. However, that means that outlier documents get a high influence on the result. Therefore, we propose to define that t is only considered element of a class C if at least X percent of the documents d contain the term - where X is a user-defined parameter (see equation 3).

$$icf(t) = log(\frac{\#classes}{|\{C \in classes : \frac{|\{d \in C : t \in d\}|}{|\{d \in C\}|} > X\}|}) \tag{3}$$

The icf value plays an important role in filtering stopwords (in the broadest sense, see section 4.2). This is due to the fact that it becomes 0 if all classes are considered as containing the term and in this case the term cannot be considered as being discriminating for any class.

The distribution of a term over the documents of a class also can reveal something about its importance for the class. There are a number of possible distribution weights that could be included into the multiplication - even several at once. We made good experiences using the standard deviation of a term's frequency as such a distribution weight but the χ^2 significance value may be used as well (both are suggested in [4] as term weights).
Another valuable choice can be the integration of a term relevance weight which was defined by Salton & Buckley [20] as "the proportion of relevant documents in which a term occurs divided by the proportion of nonrelevant items in which the term occurs". In contrast to a typical information retrieval task where the division into relevant and nonrelevant documents is not given apriori, the term relevance weight can easily be evaluated here. To calculate the term relevance weight for a term t and a class C, we simply consider all documents out of C as relevant and all documents contained by the other classes as nonrelevant. The higher the percentage of documents in C containing t and the lower the corresponding percentage for the other classes, the higher is our weight (see equation 4 and 5).

$$term\_relevance\_weight(t_i, C_j) = \frac{support(t_i, C_j)}{\sum_{k \neq j} support(t_i, C_k)} \tag{4}$$

with

$$support(t_x, C_y) = \frac{\left|\left\{D_z \in C_y : t_x \in D_z\right\}\right|}{\left|\left\{D \in C_y\right\}\right|} \tag{5}$$

3.2 Determining discriminating and overlap terms

The weighted tficf measure provides a term score that is comparable among several classes. So the next logical step is to use it for comparison. For any term we get as many scores as there are classes: For each individual class, there is a particular score. We now define that a term is discriminating for one of these classes if its score is much higher for this class than its scores for the other classes. To determine the discriminating terms for a class, we use a threshold called discrimination factor by which a score for one class must outnumber the scores for all other classes (see definition 1).

Definition 1. *Discriminating terms*

A term t is discriminating for a single class C_k if:

$\forall i \in \{1 \cdots n\} \setminus k$:
$weighted\_tficf(t, C_k) > discrimination\text{-}factor \cdot weighted\_tficf(t, C_i)$.

The same approach can be applied to determine if a term is discriminating for the overlap of several classes. This is precisely the case if the lowest term score for one of the overlap classes outnumbers the highest term score of the remaining classes at least by the threshold factor (see definition 2).

Definition 2. *Overlap terms*

For the overlap area of several classes $\{C_k, C_l, \cdots, C_m\}$ a term t is discriminating if:

$\forall i \in \{1 \cdots n\} \setminus \{k, l, \cdots, m\}$:
$min(weighted\_tficf(t, C_k), weighted\_tficf(t, C_l), \cdots, weighted\_tficf(t, C_m))$
$> discrimination\text{-}factor \cdot weighted\_tficf(t, C_i)$.

In practice both discriminating terms and overlap terms can be determined in a single scan through the database.

3.3 Parameter tuning

Our algorithm for determining the discriminating and overlapping terms has two parameters: a minimum percentage and the discrimination factor. The minimum percentage is used to specify the minimum number of documents of a class that must contain the term to allow it to be chosen as discriminative. Without that parameter all terms that only occur in one class would most certainly be considered as being discriminative even if they only occur once in that class (because $X > 0$ * factor would always be true, no matter how small the value of X is).
While the minimum percentage can easily be set by the user (e.g. 0.2 if at least 20% of the documents shall contain a term), the discrimination factor threshold cannot easily be fixed without prior experience. In our experiments reasonable thresholds showed to lie typically in the interval between 1.5 and 3.0. In our implementation the exact threshold is set by using a dynamic slider, which allows the user to get the desired amount of discriminating terms.

3.4 Preprocessing

Like in many text mining applications careful preprocessing is valuable. In our case we applied a base form reduction algorithm [21] to all words in order to get singular forms for nouns and infinitive forms for verbs. In addition we used a POS-tagger ([22], [23], [24]) and a NP-chunker ([25], [26]) to identify nouns respectively noun phrases. This allows us to focus only on nouns and noun phrases if this is desired. Numbers and short strings with less than 3 characters were deleted in the preprocessing step, since they often correspond to punctuation marks or special characters that do not need to be considered.
One interesting advantage of our method is that we do NOT use any stopword lists. High-frequent stopwords like “the” or “and” are ignored with very high probability because their icf values become 0. Stopwords with a lower frequency in a regular case should not appear considerably more often in one class than in the others and thus are filtered out.

4 Application Examples

As mentioned in section 1 of this paper, our method can be used to explore the characteristics of predefined classes. The extracted discriminating and overlap terms enable users to gain insight into the hidden underlying topical structure of sets of document classes.

4.1 Characteristic Terms for Conferences

One concrete example for the application of our method could be motivated by the questions: If we take different conferences in the computer science area, can we detect automatically by processing all of the papers published in these conferences: (a) How they differ from each other? (b) What single conferences focus on or what makes them

special? (c) What several conferences have in common, respectively what distinguishes them from the other conferences?
We tried to answer these questions for a set of 9 different conferences by regarding about 100 recently published papers for each of these conferences. Besides the NLDB conference we decided to focus on other conferences that we know well, dealing with:

- Information Retrieval (SIGIR),
- Database and Data Storage (VLDB and SIGMOD),
- Knowledge Discovery and Data Mining (KDD),
- Visual Analytics (VAST),
- Information Visualization (InfoVis),
- Visualization (VIS), and
- Computer Graphics (SIGGRAPH).

The results of our approach can be found in figure 2:

As can be seen the discriminating terms of the NLDB relate very much to natural language. Database-related vocabulary does not appear in the list as it is also covered by other conferences and thus not discriminating for NLDB in this context. NLDB has a discriminating overlap with SIGIR conference, because only those two deal with query terms and corpora. In contrast, everything related to information or document retrieval apparently is significantly more covered by the papers of the SIGIR conference. NLDB has also small discriminating overlaps with VLDB and VAST but there is no overlap with the conferences that focus on visualization and computer graphics.
Also the extracted terms for overlaps between two or more conferences fit nicely and are reasonable: E.g. SIGGRAPH and VIS share a lot of computer graphics vocabulary, and InfoVis and VAST the topic of visualizing information. SIGGRAPH, VIS and InfoVis still share some vocabulary related to graphical representations, while VIS, InfoVis and VAST all deal with visualizations. Finally, while SIGMOD and VLDB are both database conferences that share many database-related topics our method reveals that there are also differences in topic coverage. The term “database management”, for example, only occurs in the SIGMOD term list, while VLDB papers seem to focus more on topics such as “memory usage”.
One nice particularity of our method is that if a term is important for every class then it is not extracted: Although NLDB surely shares topic terms such as e.g. “algorithm” or “data” with the visualization conferences, they are not extracted, as all the other considered conferences also contain these topics. Within the context of these specific other conferences such terms are not of interest as they do not provide any discrimination power. Another interesting issue is that some proper names appear in result sets. This is an indication that certain persons and institutions seem to have strong influences on specific conferences.

4.2 Characteristic Terms in Customer Reviews (Amazon)

In a different project we worked on a data set of printer reviews from amazon.com. We were interested in the attributes that the customers frequently commented on (such as the paper tray of the printer, the cartridges etc.). However, in those reviews not only

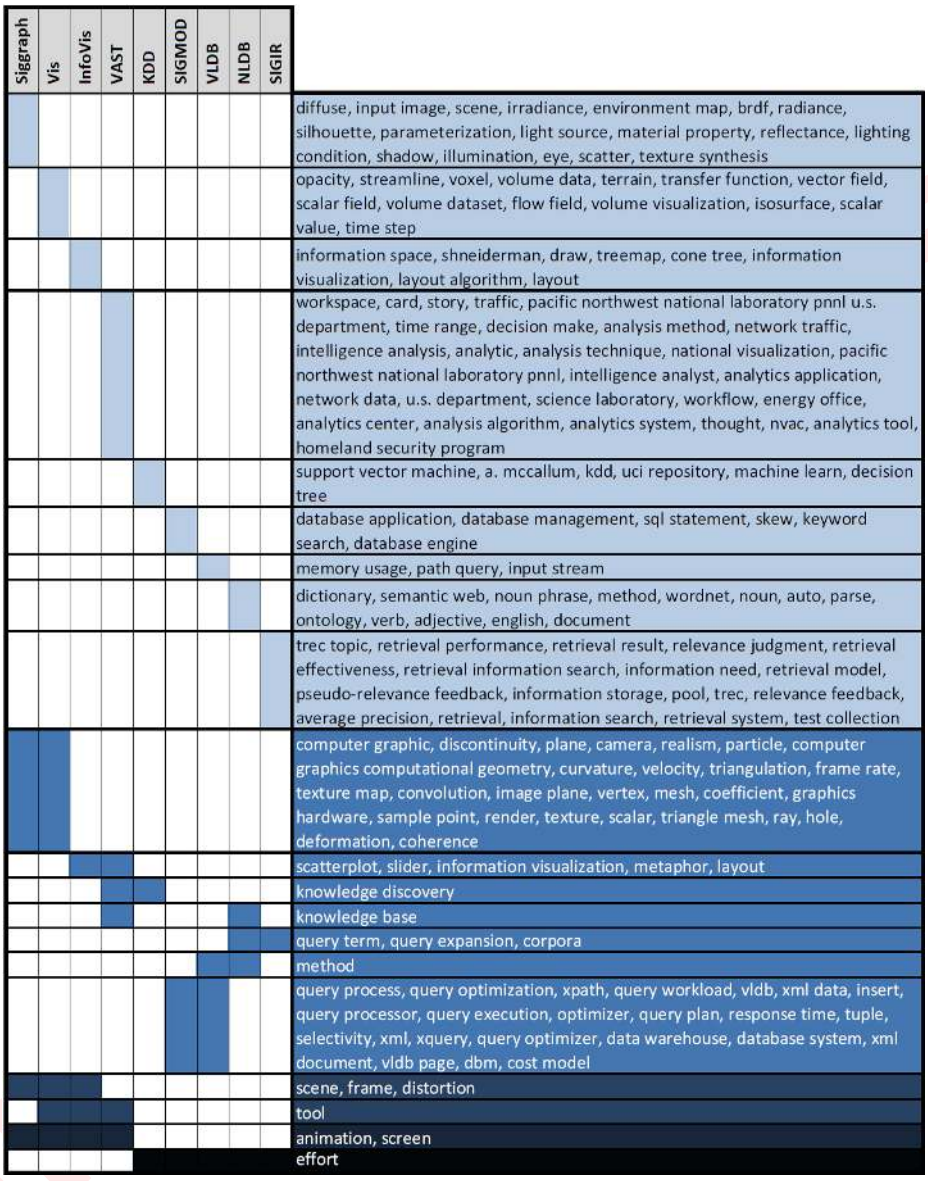

Fig. 2: On the left side the set of conferences is listed for which a set of terms is discriminating. A conference is contained in this set if its corresponding matrix entry is marked in a blue color tone. The more conferences a set contains, the darker is the blue. The corresponding terms can be found on the right side. The combinations of conferences that do not appear, simply do not jointly discriminate against the others in a certain topic.

the terms describing printer attributes occur frequently, but also the review-related vocabulary. Widely used stopword lists contain only very general terms like conjunctions, determiners, pronouns etc. and thus were not suitable to separate the printer terms from the rest. We had to apply a special term filtering that extracted the printer terms while it did not consider the review terms. For this purpose we applied our discrimination-based term extracting method: We used a counter-balance class containing book reviews and discriminated the printer review class against it. As both classes shared the review specific terms, only printer related terms were discriminating the printer class and hence got extracted. Figure 3 compares a simple approach that just extracts the 40 most frequent terms after filtering stopwords out (top) with the result of our technique using the book reviews as a counter-balance class (bottom). It is easy to see that the quality of the second list is much higher since lots of review-related terms such as "good", "like" or "need" that are uninteresting in our case are not contained in the list.

As you can see, besides getting deeper insight into the commonalities and differences of document collections our approach also allows us to do domain-specific term filtering without the usage of an ontology or a specialized knowledge base. To apply the technique a set of documents has to be provided that contains the words that we would like to be filtered out but does not contain (or does less often contain) the type of words that we are interested in. Our method is then used to extract the terms that discriminate the class of documents that we are interested in from this counter-balance class. As only terms are selected as discriminating terms that are significantly more important for one class than for the other, the aspects that the documents of both classes share are automatically filtered out. Sometimes it can be helpful to use more than one class as a counter-balance class. This is the case when there are several undesired aspects to be filtered out and there does not exist a single counter-balance class that contains all of those aspects.

40 terms with highest frequencies (stopwords have been removed):

printer, print, use, good, work, scan, buy, problem, install, software, great, time, easy, like, need, try, machine, ink cartridge, fax, ink, set, purchase, make, hp printer, copy, paper, run, product, come, price, look, say, want, photo, new, quality, real, page, wireless, think

40 discriminating terms:

network, product, ink cartridge, fax, jam, paper, scan, print quality, print, download, printer, cartridge, software, mac, unit, function, month, all-in-one, installation, machine, scanner, install, box, model, use, hp, feature, replace, easy, black, document, fix, support, driver, ink, color, wireless, photo, expensive, hp printer

Fig. 3: 40 most frequent terms (top) compared to the Top-40 discriminating terms. It can easily be seen that the list of discriminating terms is more dense with respect to the question what the customers frequently comment on while the list of the most frequent terms also contains many terms that are typically used in reviews but do not convey the desired information (e.g. need, like, good, etc).

5 Experimental evaluation

To evaluate how well the extracted terms are able to discriminate one class of documents from the others we used the extracted terms in a classification task. This was done as follows: Given three different classes of documents we used 4 different methods to extract (in average) 15 terms per class (the different methods are described in detail below). As classes we used the three conferences InfoVis, Siggraph, and Vis and each class was made up of 100 papers of the conference. The extracted terms were then used to classify a set of 60 test documents (20 of each class) that were different from the training set. Each of the 60 documents was assigned to the class that it shared most discriminating terms with. If there was more than one winning class the document was assigned to the class whose absolute frequency was largest (counting all the occurrences of discriminating terms instead of just every term once). If the document still could not be assigned unambiguously it was assigned to the class of ambiguous documents. In that classification task a method performs best if it extracts terms that discriminate a class from the others but yet also chooses terms that are characteristic for the class they have been extracted for (i.e. that they are shared by many documents of the specific class instead of being only significant for a small subset of documents of the class).

We used the following four methods for term extraction:

- TFIDF average: Given the training corpus of 300 documents for each document and each term in the corpus a TFIDF value was calculated. Afterwards the documents were sorted into classes and for each class the average TFIDF of each term was calculated. Next, the terms were sorted according to their average value. Finally, for each class the 15 top terms were chosen.
- TFIDF max: The second method is very similar to the first one. The only difference is that instead of calculating the average TFIDF value the maximum TFIDF value of the class is chosen for each term. Then, again the terms are sorted according to their TFIDF values and the 15 top terms for each class were chosen. We included this method, too, since it has been proposed in other publications ([4], [27]).
- Differential Analysis: This is a general technique that extracts technical terms from a corpus by comparing the probability of occurrence in the given corpus to a general reference corpus [9]. We used the authors online tool to extract the terms for our paper [28]. There are two main differences to our method: First, instead of comparing the different classes against each other a general reference corpus is used. Secondly, a different term weighting approach is used. As before, for each class we extracted the top 15 terms.
- Our approach: To extract terms with the approach that is proposed in this paper we set the parameter values as follows: The minimum percentage was set to 0.11 (that means that more than 10% of the documents have to contain the term) and the discrimination factor to 2.0. Since our method does not extract a given number of terms but automatically determines the number of terms that well discriminate one class from the others we do not have exactly 15 terms per class but 14 terms for InfoVis, 15 for Vis and 16 for Siggraph.

ground truth

predicted

TFIDF avg	InfoVis	Siggraph	Vis
InfoVis	19	1	4
Siggraph	1	18	9
Vis	0	1	3
ambiguous	0	0	4

TFIDF max	InfoVis	Siggraph	Vis
InfoVis	3	1	2
Siggraph	0	3	0
Vis	0	0	4
ambiguous	17	16	14

diff. analysis	InfoVis	Siggraph	Vis
InfoVis	20	1	10
Siggraph	0	18	1
Vis	0	1	8
ambiguous	0	0	1

Our approach	InfoVis	Siggraph	Vis
InfoVis	16	0	1
Siggraph	1	18	1
Vis	0	2	16
ambiguous	3	0	2

Fig. 4: Confusion matrices for the four different methods classifying 60 documents.

The evaluation result: The following accuracy values were calculated for the four methods (accuracy = number of correctly classified documents divided by the total number of documents)[1]: TFIDF avg: 0.71, TFIDF max: 0.77, Diff. analysis: 0.78, Our approach: 0.91.

Figure 4 shows the result in more detail. The large number of documents in the class "ambiguous" shows that the relatively good result for TFIDF max is misleading. Almost 80% of the documents could not be classified unambiguously. The results for the other 3 techniques are more meaningful. It can easily be seen in the confusion matrix that all the methods performed well on the classes InfoVis and Siggraph but that TFIDF avg and the Differential Analysis had problems with the class Vis. An explanation for that might be that the Vis conference is thematically somehow in between the two other conferences. The closer the classes are related to each other the more important it is that the applied method is able to find terms that are really discriminating and not only characteristic for the class as our method does.

In order to get some deeper insight we conducted a more extensive evaluation where we also analyzed the distribution of the extracted terms visually. The left graphic of figure 5 shows the distribution across the documents of the class that the terms were extracted for (we used the terms and documents of class InfoVis). The height of each bar in the graphic represents the number of documents in the training corpus that contain k extracted terms. Obviously, the distribution for TFIDF max falls apart. More than 90% of the documents contain only 1 or even 0 of the extracted terms! That means that the method extracts many terms that can only be found in very few documents of the class (which means that they cannot be considered as characteristic for the class). The three other methods show distributions that are similar to each other. The right graphic of figure 5 reveals the difference between those three methods. This time not only the distribution of the terms across the class that the terms were extracted for has been analyzed but also the distribution across the two other classes. As can clearly be seen our approach is the only one that favors terms that often occur in the corresponding class but rarely in other classes.

[1] Ambiguous documents were ignored in the accuracy calculation.

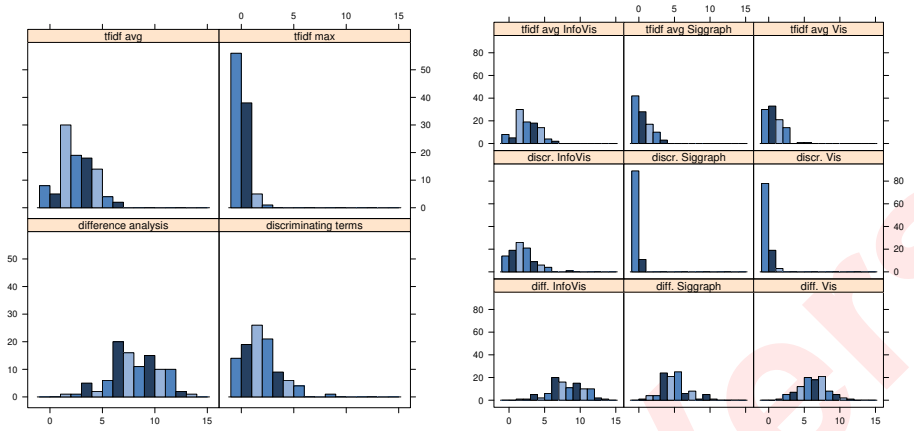

Fig. 5: Analysis of the distribution of the terms, comparing the three methods TFIDF avg, TFIDF max and Differential Analysis to our method (Discriminating Terms). Left: Distribution across the documents of the class that the terms were extracted for (InfoVis). The height of each bar in the graphic represents the number of documents in the training corpus that contain k extracted terms (with k being mapped to the x-axis). Right: Distribution across the documents of the other two classes that the terms were *not* extracted for.

6 Conclusion

In this paper we presented a novel approach for the extraction of discriminating and overlap terms out of a set of document classes. By applying our method to two important application scenarios, we were able to demonstrate its relevance and performance for real problems.

First, our method gives insight into the topical coherences and differences among several distinct document classes, e.g. the papers of scientific conferences. Secondly, our method allows us to do domain-specific term filtering. The discrimination calculation is able to filter out automatically the vocabulary that covers a certain aspect or has a certain function. With both applications we are able to show that our method not only yields very good results but also can be applied easily and in a very flexible way.

While we apply some language dependent preprocessing techniques like base form reduction, POS tagging and NP chunking, the core of our approach is language independent.

Finally, we evaluate our method using the extracted terms in a classification task, where it yields better results than a number of other methods for term extraction. We assure that our method extracts discriminating terms that at the same time have a high relevance for a class.

A wider range of promising application scenarios is easily imaginable - wherever classes

of documents differ in topical, stylistic or linguistic features, and those differences on their part are of interest.

Acknowledgment This work has partly been funded by the Research Initiative "Computational Analysis of Linguistic Development" at the University of Konstanz and by the German Research Society (DFG) under the grant GK-1042, Explorative Analysis and Visualization of Large Information Spaces, Konstanz.
We thank the anonymous reviewers of the NLDB 2009 for their valuable comments.

References

1. Spärck Jones K.: A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation. **28:1** (1972) 11–21
2. Salton G., Wong A., Yang, C.S.: A Vector Space Model for Automatic Indexing. Communications of the ACM. **18:11** (1975) 613–620
3. Kageura, K., Umino B.: Methods of automatic term recognition: A review. Terminology **3:2** (1996) 259ff
4. Feldman R., Fresko M., Kinar Y., Lindell Y., Liphstat O., Rajman M., Schler Y, Zamir O.: Text mining at the term level. Proceedings of the Second European Symposium on Principles of Data Mining and Knowledge Discovery. (1998) 65–73
5. Matsuo Y., Ishizuka M.: Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information. Proceedings of the 16th International Florida AI Research Society. (2003) 392–396.
6. Riloff E., Jones R.: Learning dictionaries for information extraction by multi-level bootstrapping. Proceedings of the Sixteenth National Conference on Artificial Intelligence. (1999) 474–479
7. Collins M., Singer Y.: Unsupervised models for named entity classification. Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora. (1999)
8. Brunzel M., Spiliopoulou M.: Domain Relevance on Term Weighting. 12th International Conference on Applications of Natural Language to Information Systems. (2007) 427-432
9. Witschel H.F.: Terminologie-Extraktion: Möglichkeiten der Kombination statistischer und musterbasierter Verfahren. Content and Communication: Terminology, Language Resources and Semantic Interoperability. Ergon Verlag, Würzburg. (2004)
10. Velardi P., Missikoff M., Basili R.: Identification of relevant terms to support the construction of domain ontologies. Proceedings of the workshop on Human Language Technology and Knowledge Management. (2001) 1–8
11. Drouin P.: Detection of Domain Specifc Terminology Using Corpora Comparison. Proceedings of the International Language Resources Conference. (2004) 79–82
12. Wise J.A.: The ecological approach to text visualization. Journal of the American Society for Information Science. (1999) 1224–1233
13. Kaski S., Honkela T., Lagus K., Kohonen T.: WEBSOM Selforganizing maps of document collections. Neurocomputing vol. 21. (1998) 101-117
14. Lagus K., Kaski S.: Keyword selection method for characterizing text document maps. Proceedings of ICANN99, Ninth International Conference on Artificial Neural Networks. (1999) 371–376
15. Azcarraga A.P., Yap T.N., Tan J., Chua T.S.: Evaluating Keyword Selection Methods for WEBSOM Text Archives. IEEE Transactions on Knowledge and Data Engineering. **16:3** (2004) 380–383

16. Seki Y., Eguchi K., Kando N.: Multi-Document Viewpoint Summarization Focused on Facts, Opinion and Knowledge. In Computing Attitude and Affect in Text: Theory and Applications (The Information Retrieval Series), Springer. (2005) 317–336.
17. Lerman K., McDonald R.: Contrastive Summarization: An Experiment with Consumer Reviews. Proceedings of the North American Association for Computational Linguistics (NAACL). (2009)
18. Zhai C., Velivelli A., Yu B.: A Cross-Collection Mixture Model for Comparative Text Mining. In Proceedings of the ACM SIGKDD international conference on Knowledge discovery and data mining (KDD). (2004) 743–748
19. Mei Q., Zhai C.: A mixture model for contextual text mining. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD). (2006) 649–655
20. Salton G., Buckley C.: Term weighting approaches in automatic text retrieval. Information Processing and Management: an International Journal. **24:5** (1988) 513–523
21. Kuhlen R.: Experimentelle Morphologie in der Informationswissenschaft. Verlag Dokumentation. (1977)
22. Toutanova K., Manning C.: Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC). (2000) 63–70
23. Toutanova K., Klein D., Manning C., Singer Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL). (2003) 173–180
24. Stanford Log-linear Part-Of-Speech Tagger, http://nlp.stanford.edu/software/tagger.shtml
25. Ramshaw L., Marcus M.: Text Chunking Using Transformation-Based Learning. In Proceedings of the Third ACL Workshop on Very Large Corpora. (1995)
26. Greenwood M.: Noun Phrase Chunker Version 1.1, http://www.dcs.shef.ac.uk/ mark/phd/software/chunker.html
27. Thiel K., Dill F., Kötter T., Berthold M.R.: Towards Visual Exploration of Topic Shifts. IEEE International Conference on Systems, Man and Cybernetics. (2007) 522–527
28. Online tool for terminology extraction: http://wortschatz.uni-leipzig.de/ fwitschel/terminology.html