Unsupervised and domain-independent extraction of technical terms from scientific articles in digital libraries

Kathrin Eichler, Holmer Hemsen and Günter Neumann
DFKI Project Office Berlin
Alt-Moabit 91c, Berlin
{kathrin.eichler, holmer.hemsen, neumann}@dfki.de

Abstract

A central issue for making the contents of documents in a digital library accessible to the user is the identification and extraction of technical terms. We propose a method to approach this task in an unsupervised, domain-independent way: We use a nominal group chunker to extract term candidates and select the technical terms from these candidates based on string frequencies retrieved using the MSN search engine.

1 Introduction

Digital libraries (DL) for scientific articles are more and more commonly used for scientific research. Prominent examples are the Association for Computing Machinery digital library or the Association for Computational Linguistics anthology. DL may easily contain several millions of documents, especially if the DL covers various domains, such as Google Scholar. The content of these documents needs to be made accessible to the user in such a way that the user is assisted in finding the information she is looking for. Therefore, providing the user with sufficient search capabilities and efficient ways of inspecting the search results is crucial for the success of a digital library. Current DL often restrict the search to a small set of meta-labels associated with the document, such as title, author names, and keywords defined by the authors. This restricted information may not be sufficient for retrieving the documents that are most relevant to a specified query.

The extraction of technical terms (TTs) can improve searching in a DL system in two ways: First, TTs can be used for clustering the documents and help the user in finding documents related to a document of interest. Second, the TTs can be provided to the user directly, in the form of a list of keywords associated with the document, and help the user in getting a general idea of what a document of interest is about. Our input documents being scientific papers, key terms of the paper can be found in the abstract. Extracting TTs from the abstract of the document only allows for an efficient processing of documents, an important issue when dealing with large amounts of data.

In this paper, we propose a method for extracting TTs in an unsupervised and domain-independent way. The paper is organized as follows. In section 2 we describe the task of technical term extraction and introduce our approach towards solving this task. After a section on related work (3), section 4 is about the generation of TT candidates based on nominal group (NG) chunking. Section 5 describes the approaches we developed to select the TTs from the list of extracted NG chunks. After a section on our experimental results (6), we describe ideas and challenges in TT categorization (7) and conclude with suggestions for future work in section 8.

2 Technical term extraction

The task of extracting technical terms (TTs) from scientific documents can be viewed as a type of Generalized Name (GN) recognition, the identification of single- or multi-word domain-specific expressions [Yangarber *et al.*, 2002]. Compared to the extraction of Named Entities (NEs), such as person, location or organization names, which has been studied extensively in the literature, the extraction of GNs is more difficult for the following reasons: For many GNs, cues such as capitalization or contextual information, e.g. 'Mr.' for person names or 'the president of' for country names, do not exist. Also, GNs can be (very long) multi-words (e.g. the term 'glycosyl phosphatidyl inositol (GPI) membrane anchored protein'), which complicates the recognition of GN boundaries. An additional difficulty with domain-independent term extraction is that the GN types cannot be specified in advance because they are highly dependent on the domain. Also, we cannot make use of a supervised approach based on an annotated corpus because these corpora are only available for specific domains. Our idea for domain-independent term extraction is based on the assumption that, regardless of the domain we are dealing with, the majority of the TTs in a document are in nominal group positions. To verify this assumption, we manually annotated a set of 100 abstracts from the *Zeitschrift für Naturforschung*[1] (ZfN) archive. Our complete ZfN corpus consists of 4130 abstracts from scientific papers in physics, chemistry, and biology, published by the ZfN between 1997 and 2003. Evaluating 100 manually abstracts from the biology part of the ZfN corpus, we found that 94% of the annotated terms were in fact in noun group (NG) positions. The remaining 6% include TTs in verb positions (e.g. 'debug' in the sentence 'Reprogrammable hardware systems are traditionally very difficult to debug'), but also terms occurring within an NG, where the head of the NG is not part of the TT. For example, in the NG 'Java program', the head of the noun group ('program') is not part of the TT ('Java'). Focussing our efforts on the terms in NG position, the starting point of our method for extracting terms is an algorithm to extract nominal groups from a text. We then classify these nominal groups into TTs and non-TTs using frequency counts retrieved from the MSN search engine.

[1] http://www.znaturforsch.com/

3 Related work

NE and GN extraction tasks have long been tackled using supervised approaches. Supervised approaches to standard NE extraction tasks (person, organization, location, etc.) have been discussed in various papers, e.g. [Borthwick *et al.*, 1998] and [Bikel *et al.*, 1999]. A supervised (SVM-based) approach to the extraction of GNs in the biomedical domain is presented by [Lee *et al.*, 2003]. As a major drawback of supervised methods is the need for manually-tagged training data, people have, during the last decade, looked for alternative approaches. Lately, bootstrapping has become a popular technique, where seed lists are used to automatically annotate a small set of training samples, from which rules and new instances are learned iteratively. Seed-based approaches to the task of learning NEs were presented by, e.g. [Collins and Singer, 1999], [Cucerzan and Yarowsky, 1999], and [Riloff and Jones, 1999]. [Yangarber *et al.*, 2002] present a seed-based bootstrapping algorithm for learning GNs and achieve a precision of about 65% at 70% recall, evaluating it on the extraction of diseases and locations from a medical corpus. Albeit independent of annotated training data, seed-based algorithms heavily rely on the quality (and quantity) of the seeds. For extracting GNs in a completely domain-independent way, these lists of trusted seeds are simply not available and can hardly be generated automatically. A different approach, which does not rely on seeds, is applied by [Etzioni *et al.*, 2005], who use [Hearst, 1992]'s list of lexico-syntactic patterns (plus some additional patterns) to extract NEs from the web. The patterns are extended with a predicate specifying a class (e.g. City) to extract instances of this particular class. The extracted instances are validated using an adapted form of [Turney, 2001]'s PMI-IR algorithm (point-wise mutual information). This allows for a domain-independent extraction of NEs but only from a huge corpus like the internet, where a sufficient number of instances of a particular pattern can be found. Also, using this approach, one can only extract instances of categories that have been specified in advance.

4 NG Chunking

For the extraction of term candidates, we use the nominal group (NG) chunker of the GNR tool developed by [Spurk, 2006]. The advantage of this chunker compared to other chunkers is its domain-independence. This is due to the fact that it is not trained on a particular corpus but relies on patterns based on closed class words (e.g. prepositions, determiners, coordinators), which are the same in all domains. Using lists of closed-class words, the NG chunker determines the left and right boundaries of a word group and defines all words in between as an NG. However, the boundaries of a term do not always coincide with the boundaries of a nominal group. For example, from the nominal group 'the amino acid', we want to extract the term 'amino acid'. Therefore, we made some adaptations to the chunker in order to eliminate certain kinds of pre-modifiers. In particular, we made the chunker to strip determiners, adverbs, pronouns and numerals from the beginning of an NG. We also split coordinated phrases into their conjuncts, in particular comma-separated lists, and process the text within parentheses separately from the text outside the parentheses.

5 Selection of Technical Terms

5.1 Seed-based approach

Our first approach towards determining which of the extracted NGs are in fact TTs was to use Wikipedia for validating part of the extracted chunks (i.e. those that constitute entries in Wikipedia, about 8% of the terms in our annotated abstracts) and use these validated chunks as seeds to train a seed-based classifier. To test this approach, we used DBPedia [Auer *et al.*, 2007] (a structured representation of the Wikipedia contents) to validate the chunks and used the validated chunks as seeds for training a seed-based GN Recognizer implemented by Spurk (2006). Seed lists were generated in the following way: We first looked up all extracted NG chunks in DBPedia. For DBPedia categories, we generated a list of all instances having this category, for instances, we retrieved all categories the instance belonged to. For each category candidate, for which at least two different instances were found in our corpus, we then created a seed list for this category, containing all instances found for this category in DBPedia. For each instance candidate, we generated seed lists for each category of the instance accordingly. These lists were used as positive evidence when training the seed-based GN Recognizer. In addition, we used seed lists containing frequent words, serving as negative evidence to the learner. Our frequent word seed lists were generated from a word frequency list based on the British National Corpus. From this list, we extracted each word together with its PoS tag and frequency. After preprocessing the data (i.e. removing the '*' symbol at the end of a word and removing contractions), we generated a list of words for each PoS tag separately.

An evaluation of the seed-based GN learner on the ZfN corpus (4130 abstracts) showed that the results were not satisfying. Learning to extract instances of particular categories, the number of found sample instances in the corpus was too small for the learner to find patterns. Experiments on learning to extract instances of a general type "technical term" showed that the TTs are too diverse to share term-inherent or contextual patterns.

In particular, the use of DBPedia for the generation of seed lists turned out unpractical for the following reasons: 1. DBPedia is not structured like an ontology, i.e. instances and categories are often not in an is-a-relation but rather in an is-related-to-relation. For example, for the category 'protein', we find instances that are proteins, such as 'Globulin', but we also find instances such as 'N-terminus' that are related to the term 'protein' but do not refer to a protein. However, as the seed-based learner relies on morphological and contextual similarities among instances of the same type when trying to identify new instances, better results could only be achieved using a knowledge base, in which instances and categories are structured in a clearly hierarchical way. 2. Seed-based learning only makes sense for "open-class" categories. However, for some categories that we extracted from DBPedia, a complete (or almost complete) list of instances of this category was already available. For example, for the category 'chemical element', we find a list of all chemical elements and will hardly be able to find any new instance of this category in our input texts. In addition, we found that a number of terms that appeared as entries in DBPedia were in fact too general to be considered TTs, i.e. an entry such as 'paper'.

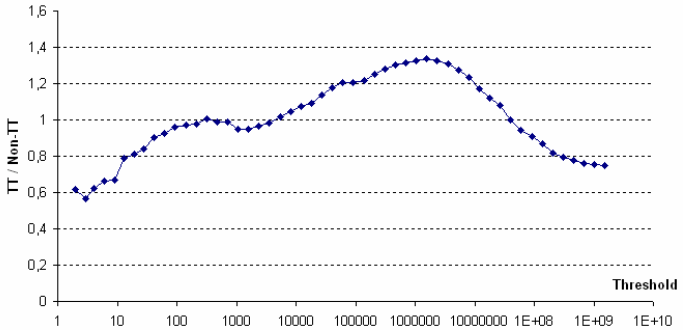

Figure 1: Ratio between TTs and non-TTs

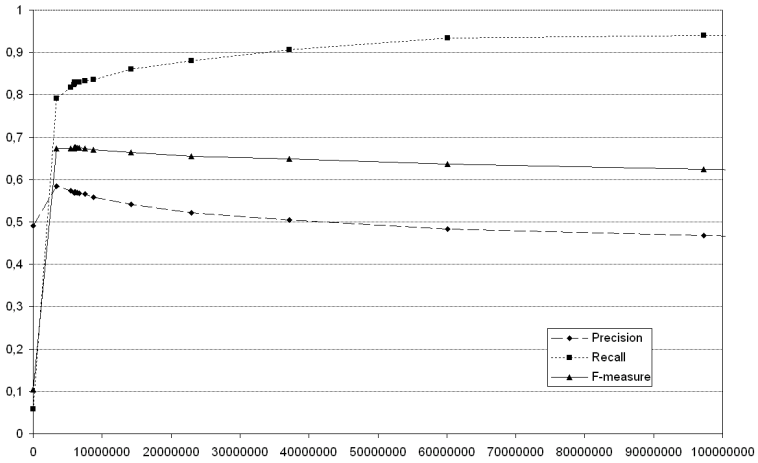

Figure 2: Determination of the optimal threshold based on F-measure maximization for the annotated ZfN corpus

5.2 Frequency-based approach

As the seed-based approach turned out unfeasible for solving the task at hand, we decided to find the TTs within the extracted NG chunks using a frequency-based approach instead. The idea is to make use of a model introduced by [Luhn, 1958], who suggested that mid-frequency terms are the ones that best indicate the topic of a document, while very common and very rare terms are less likely to be topic-relevant terms. Inspired by Luhn's findings, we make the assumption that terms that occur mid-frequently in a large corpus are the ones that are most associated with some topic and will often constitute technical terms. To test our hypothesis, we first retrieve frequency scores for all NG chunks extracted from our corpus of abstracts from the biology domain and then calculate the ratio between TTs and non-TTs for particular maximum frequency scores. To retrieve the frequency scores for our chunks, we use the internet as reference corpus, as it is general enough to cover a broad range of domains, retrieve the scores using the Live Search API of the MSN search engine[2]. The results, presented in Figure 1 on a logarithmic scale, confirm our hypothesis, showing that the ratio increases up to a MSN score of about 2 million and then slowly declines. This means that chunks with mid-frequency score are in fact more likely to be technical terms than terms with a very low or very high score.

Selecting the terms that are most likely to be technical terms requires the determination of two thresholds: the lower threshold t_l and the upper threshold t_u for classifying a term candidate c with an MSN score $msn(c)$ as TT or non-TT:

$$class(c) = \begin{cases} TT & \text{if } t_l <= msn(t) <= t_u \\ nonTT & \text{elsewhere} \end{cases} \quad (1)$$

To optimize these two thresholds, we maximized the F-measure achieved on our annotated corpus (abstracts from the biological domain of the ZfN corpus) with different thresholds set. For t_l, we simply tried all thresholds from 0 to 10 and found a threshold of 1 to yield the best results. This might seem surprising; however, as many technical terms are in fact retrieved only once or twice by MSN, recall drops dramatically very fast if a higher value of t_l is chosen. For t_u, rather than trying out all numbers from 1 to several million, we used a simple but robust optimization algorithm - golden-section search [Kiefer, 1953] - to converge towards the optimum threshold. Using this method, we determined an upper threshold of 6.05 million (cf. Figure 2) for the ZfN corpus. In order to find out whether this threshold is different for other domains, we applied the same method to optimize the threshold for a corpus from a different domain (computer science). This second corpus consists of 100 abstracts extracted from the DBLP[3] database and, like the ZfN corpus, was hand-annotated for TTs. For this corpus, the maximum F-measure was achieved with a threshold of about 20 million. We are currently developing methods for determining this threshold automatically, without using annotated training data.

6 Experimental results

Evaluating our algorithm on our two annotated corpora of abstracts, we obtained the following results, summarized

[2]http://dev.live.com/livesearch/

[3]http://www.informatik.uni-trier.de/ ley/db/

in Table 1. From the biology corpus, our NG chunker was able to extract 1264 (63.2%) of the 2001 annotated TTs in NG position completely and 560 (28.0%) partially. With the optimized threshold of 6.05 million, we achieved a precision of 57.0% at recall 82.9% of the total matches. For the DBLP data, the chunker achieved 897 (68.2%) total matches and 412 (31.3%) partial matches of the 1316 annotated NG terms. Here, with the optimized threshold of 20 million, we achieved a precision of 47.5% at recall 65.6%. The results for the ZfN corpus are comparable to results for GN learning, e.g. those by [Yangarber *et al.*, 2002] for extracting diseases from a medical corpus. For the DBLP corpus, they are considerably lower, which can be explained by the fact that terminology from the computer science domain is much more commonly found in the internet than terminology from other domains. This results in a greater overlap of TTs and non-TTs with similar MSN frequencies and, consequently, in lower classification performance.
Recall could be increased considerably if partial matches could be turned into total matches, i.e. if the correct boundaries of these TTs could be located. Partial matches are commonly due to

1. additional premodifiers, e.g. 'new iridoid glycoside' (NG) vs. *iridoid glycoside* (TT)
2. appositive constructions, e.g. 'endemic Chilean plant Latua pubiflora' (NG) vs. *Latua pubiflora* (TT)
3. extraction errors, e.g. 'induce hemolysis' (extracted) vs. *hemolysis* (TT)

A method for determining whether a premodifier is part of a TT or not is to calculate the collocation strength between premodifier and head noun and include or exclude the premodifier based on the calculated score. A similar approach can be used to split appositive constructions. To deal with extraction errors, we are currently evaluating methods to improve the TT candidate extraction component by learning domain-specific extraction patterns from the target corpus in an unsupervised way to supplement the domain-independent extraction patterns currently applied by the GNR.
Up to now, we are not able to categorize the extracted TTs, as is usually done in GN learning. However, the key advantage of our approach over other approaches to GN learning is that it extracts a broad range of different TTs robustly and irrespective of the existence of morphological or contextual patterns in a training corpus. It works independent of the domain, the length of the input text or the size of the corpus, in which in the input document appears.

	Precision	**Recall**	**F1**
ZfN (biology)	58%	81%	0,68
DBLP	48%	65%	0,55
Yangarber (diseases)	65%	70%	0,67

Table 1: Results achieved on annotated corpora

7 Categorization of technical terms

In contrast to classical named entity and GN recognition our approach does not automatically perform a categorization of the terms extracted. To avoid to implement a domain dependent solution, we have analyzed the use of DBPedia for determining categories. Every instance found in DBPedia has one or more categories associated. However, the problems of using DBPedia for categorization are

1. to identify the correct domain to which a category belongs so that the domain matches the one of the article. For example, 'vitamin C' is a song, a music group, an album but also an instance of categories from the biology domain.
2. to choose an appropriate category. Some instances belong to several categories of the same domain. For example 'vitamin C' belongs to categories 'vitamins', 'food antioxidants', 'dietary antioxidants', 'organic acids' etc. To choose an appropriate category, one approach is to also search for the category in the document and if found take this. However, this might lead to a labelling of identical instances in different documents with different categories, and as a result inconsistent clustering of documents.
3. to identify the specificity of the category. The categories found for an instance do not always have the same level of specificity. There might be categories that are more general than others or categories that have a subcategory-supercategory relationship, for example, for the instance 'strain' the categories 'microbiology' and 'microbiology terms' can be found in DBPedia. Identifying these relationships and the levels of specificity is important for clustering instances/documents.
4. to categorise instances not found in DBPedia. Last but not least, a problem in using DBPedia as source for categorizing instances is that instances might not be found in DBPedia and additional strategies for categorizing these instances need to be developed.

To deal with the first two problems, we are currently evaluating a PMI-IR-based approach. The idea is to use [Turney, 2001]'s formula to determine the best category for a given instance in a particular context. Turney computes the semantic similarity between an instance and a category in a given context by issuing queries to a search engine. The score of a particular choice (in our case: one of the categories) is determined by calculating the ratio between the hits retrieved with a problem (in our case: the instance) together with the choice and a context (in our case: other terms in the document that have already been validated) and hits retrieved with the choice and the context alone.

8 Conclusion and current challenges

We have presented a robust method for domain-independent, unsupervised extraction of TTs from scientific documents with promising results. Figure 3 shows a sample abstract extracted from a scientific paper of the CiteSeer[4] digital library. The selected TTs are shaded in yellow, the extracted term candidates are shaded in grey in the text below. The info box shows the list of DBPedia categories found for the TT 'Euler characteristic'.
Current challenges include improving the TT candidate extraction component, in particular the recognition of TT boundaries, in order to reduce the number of partial matches. For TT selection, our goal is to determine MSN frequency thresholds without using annotated training data. Another major challenge is the categorization of TTs.

[4] http://citeseer.ist.psu.edu/

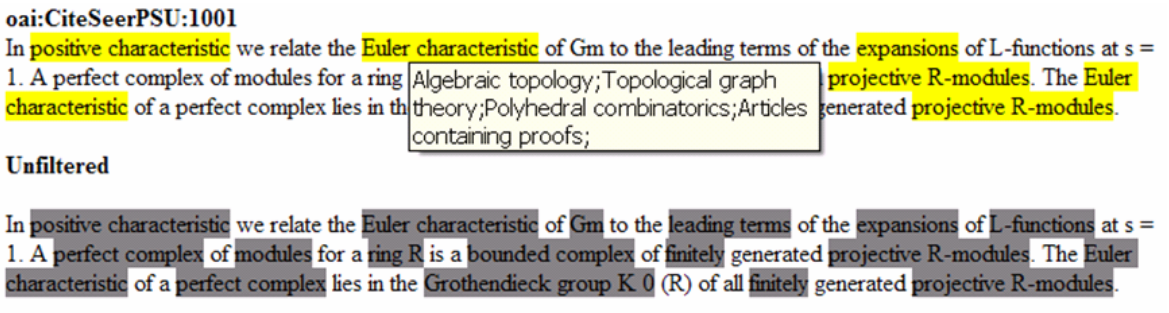

Figure 3: Sample output of our TT extraction algorithm

Acknowledgments

The project is a collaboration of the German Research Center for Artificial Intelligence GmbH (DFKI) and the Max-Planck Digital Library (MPDL). The DiLiA project is co-financed by the EFRE-programme of the European Union.

References

[Auer *et al.*, 2007] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Conference (ISWC 2007)*, Busan, Korea, November 2007.

[Bikel *et al.*, 1999] D. M. Bikel, R. Schwartz, and R. M. Weischedel. An Algorithm that Learns What's in a Name. *Machine Learning*, 34:211–231, 1999.

[Borthwick *et al.*, 1998] A. Borthwick, J. Sterling, E. Agichstein, and R. Grishman. NYU: Description of the MENE named entity system as used in MUC-7. In *Proceedings of the 7th Message Understanding Conference*, 1998.

[Collins and Singer, 1999] M. Collins and Y. Singer. Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 100–111, Maryland, USA, 1999.

[Cucerzan and Yarowsky, 1999] S. Cucerzan and D. Yarowsky. Language independent named entity recognition combining morphological and contextual evidence. In *Proceedings of the Joint SIGDAT Conference on EMNLP/VLC*, 1999.

[Etzioni *et al.*, 2005] Oren Etzioni, M. Cafarella, D. Downey, A.-M. Popescu, T. Shaked, S. Soderland, D.S. Weld, and A. Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165:91–134, 2005.

[Hearst, 1992] M. Hearst. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th International Conference on Computational Linguistics*, pages 539–545, 1992.

[Kiefer, 1953] J. Kiefer. Sequential minimax search for a maximum. In *Proceedings of the American Mathematical Society 4*, 1953.

[Lee *et al.*, 2003] K. Lee, Y. Hwang, and H. Rim. Two-phase biomedical NE recognition based on SVMs. In *Proceedings of the ACL 2003 Workshop on Natural Language Processing in Biomedicine*, 2003.

[Luhn, 1958] H.-P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2:157–165, 1958.

[Riloff and Jones, 1999] E. Riloff and R. Jones. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the 16th National Conference on AI (AAAI-99)*, Orlando, FL, 1999.

[Spurk, 2006] C. Spurk. Ein minimal berwachtes Verfahren zur Erkennung generischer Eigennamen in freien Texten. Diplomarbeit, Saarland University, Germany, 2006.

[Turney, 2001] P.D. Turney. Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the 12th European Conference on Machine Learning*, Freiburg, Germany, 2001.

[Yangarber *et al.*, 2002] R. Yangarber, L. Winston, and R. Grishman. Unsupervised learning of generalized names. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002)*, 2002.