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 solve 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, 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 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 us to process documents efficiently, 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. In section 6, we present our experimental results. We describe challenges in and first results for TT categorization (section 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 TT 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 (NG) 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 4,130 abstracts from scientific papers in physics, chemistry, and biology, published by the ZfN between 1997 and 2003. Evaluating 100 manually annotated abstracts from the biology part of the ZfN corpus, we found that 94% of the annotated terms were in fact in NG positions. The remaining 6% include TTs in verb positions, but also terms occurring within an NG, where the head of the NG is not part of the TT. For example, in the NG "Codling moth females", the head of the noun group ("females") is not part of the TT ("Codling moth"). 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

3.1 NE and GN recognition

NE and GN recognition tasks have long been tackled using supervised approaches. Supervised approaches to standard NE recognition 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]. Since 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. As lists of trusted seeds are not available for all domains, extracting GNs in a completely domain-independent way would require generating these lists automatically. A different approach, which does not rely on seeds, is applied by [Etzioni *et al.*, 2005], who use Hearst's [Hearst, 1992] 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's [Turney, 2001] 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.

3.2 Keyword extraction

The goal of keyword extraction from a document is to extract a set of terms that best describe the content of the document. This task is closely related to our task; however, we aim at extracting *all* TTs rather than a subset. Like NE/GN recognition, keyword extraction was first approached with supervised learning methods, e.g. [Turney, 2000] and [Hulth, 2003]. [Mihalcea and Tarau, 2004] propose to build a graph of lexical units that are connected based on their co-occurrence and report an F-measure of 36.2 on a collection of manually annotated abstracts from the Inspec database. [Mihalcea and Csomai, 2007] identify important concepts in a text relying on Wikipedia as a resource and achieve an F-measure of 54.63. However, limiting the extracted concepts to those found in Wikipedia is problematic when working on specialized texts. Evaluating the annotated technical terms of the GENIA (Technical Term) Corpus, an annotated corpus of 2000 biomedical abstracts from the University of Tokyo[2], we found that only about 15% of all annotated terms (5.199 out of 34.077) matched entries in Wikipedia.

[2]http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/

4 NG chunking

As TTs are usually in noun group positions, we extract candidates using a nominal group (NG) chunker, namely the GNR chunker developed by [Spurk, 2006]. The advantage of this chunker over others is its domain-independence, 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 TT do not always coincide with the boundaries of an NG. For example, from the NG "the amino acid", we want to extract the TT "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.
Evaluating the NG chunker for TT candidate extraction, we ran the chunker on two sets of annotated abstracts from the biology domain (ZfN and GENIA) and a set of 100 abstracts extracted from the DBLP[3] database (computer science), which was hand-annotated for TTs. To evaluate the chunker on the GENIA data, we first had to identify the annotated terms in NG position. Considering all terms with PoS tags[4] matching the regular expression $JJ^*NN^*(NN|NNS)$ as NG terms, we extracted 62.4% of all terms (57,845 of 92,722). Table 1 shows the performance of the NG chunking component of our system, evaluated on the annotated TTs in NG position of the three corpora.

	NG TTs	**total matches**	**partial matches**
ZfN	2,001	1,264 (63.2%)	560 (28.0%)
DBLP	1,316	897 (68.2%)	412 (31.3%)
GENIA	57,845	45,660 (78.9%	10,321 (11.9%)

Table 1: Evaluation of NG chunking on annotated corpora

The high number of partial matches in all corpora might be surprising; however, in many cases, these partial matches, even though untagged by the annotator, constitute acceptable TT candidates themselves. Some are due to minor variances between manual annotation and chunking, e.g. a missing dot at the end of the TT "Ficaria verna Huds." in the chunking output, or due to the fact that the extracted NG chunk is a super- or sub-NG of the annotated NG term. Common causes for partial matches are:

1. missing prepositional postmodifier, e.g. "biodegradation" and "Naphthalene" (NGs) vs. "Biodegradation of Naphthalene" (TT)
2. additional premodifiers, e.g. "new iridoid glycoside" (NG) vs. "iridoid glycoside" (TT)
3. appositive constructions, e.g. "endemic Chilean plant Latua pubiflora" (NG) vs. "Latua pubiflora" (TT)

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

[4]PoS tag annotation follows the Penn Treebank tagging scheme

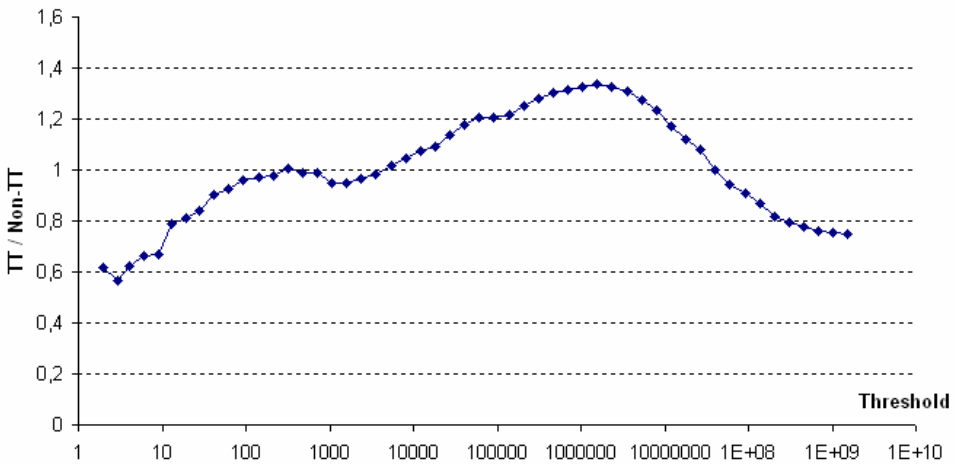

Figure 1: Ratio between TTs and non-TTs (ZfN corpus)

Real chunking errors are usually due to leading or trailing verbs, e.g. “induce hemolysis” (extracted) vs. “hemolysis” (TT). To deal with these 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.

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[5]. 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 (4,130 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”.

5.2 Frequency-based approach

As the seed-based approach turned out unfeasible for solving the task at hand, we decided to identify 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 retrieved frequency scores for all NG chunks extracted from our ZfN corpus of abstracts from the biology domain and then calculated the ratio between TTs and non-TTs for particular maximum frequency

[5] http://www.natcorp.ox.ac.uk/

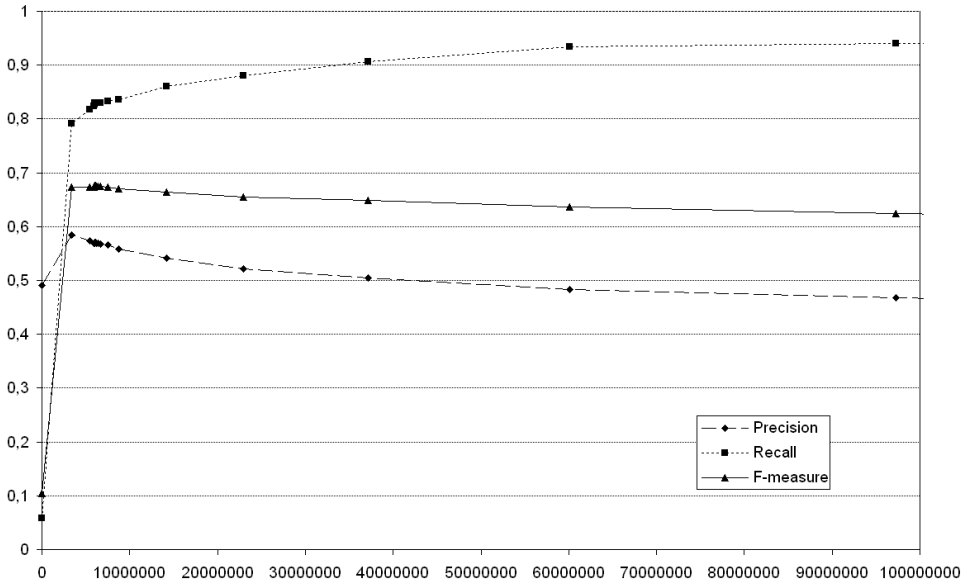

Figure 2: Optimization of t_u based on F-measure maximization (ZfN corpus)

scores. To retrieve the frequency scores for our chunks, we used the internet as reference corpus, as it is general enough to cover a broad range of domains, and retrieved the scores using the Live Search API of the MSN search engine[6]. The results, presented in Figure 1 on a logarithmic scale, confirm our hypothesis, showing that the ratio increases up to an MSN score of about 1.5 million and then slowly declines. This means that chunks with a mid-frequency score are in fact more likely to be TTs than terms with a very low or very high score.
Selecting the terms that are most likely to be TTs 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:

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

To optimize these two thresholds, we maximized the F-measure achieved on 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 a (local) 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 the DBLP corpus (computer science). 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]http://dev.live.com/livesearch/

6 Experimental results

Evaluating our algorithm on our three annotated corpora of abstracts, we obtained the results summarized in Table 2. The scores 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.
To evaluate our approach in an unsupervised way (i.e. without using the annotated corpora for threshold optimization), we selected the top half[7] of the extracted NG chunks as TTs and compared this list to the set of annotated TTs and to a set of the top half of extracted NG chunks selected using TF/IDF, a baseline measure commonly applied in keyword extraction. As "top half", we considered the chunks with the lowest MSN score (with an MSN score of at least 1) and those chunks with the highest TF/IDF score, respectively. The results, summarized in Table 3, show that our MSN-based method yields considerably better results than the TF/IDF baseline. The F-measure of 0.55 for terms in NG position corresponds to the score achieved by [Mihalcea and Csomai, 2007] for Wikipedia terms. However, our method does not limit the extracted terms to those appearing as entries in the Wikipedia encyclopedia.
Figure 3 shows a sample abstract from the ZfN corpus, with the identified TTs shaded.

7 Categorization of technical terms

In contrast to classical NE and GN recognition, our approach does not automatically perform a categorization of the extracted terms. For a domain-independent approach towards categorization, we have analyzed the use of DBpedia. Every instance found in DBpedia has one or more

[7]Analysing our different corpora, we found that the number of TTs annotated in a text is usually about half the number of extracted NGs

Acid phosphatase activities in a culture liquid and mycelial extract were studied in submerged cultures of the filamentous fungus Humicola lutea 120-5 in casein-containing media with and without inorganic phosphate (Pi). The Pi-repressible influence on the phosphatase formation was demonstrated. Significant changes in the distribution of acid phosphatase between the mycelial extract and culture liquid were observed at the transition of the strainfrom exponential to stationary phase. Some differences in the cytochemical localization of phosphatase in dependence of Pi in the media and the role of the enzyme in the release of available phosphorus from the phosphoprotein casein for fungal growth were discussed.

Figure 3: ZfN sample output of the TT extraction algorithm

	Precision	**Recall**	**F1**
ZfN (biology)	58%	81%	0.68
DBLP (computer science)	48%	65%	0.55
GENIA (biology)	50%	75%	0.60
Yangarber (diseases)	65%	70%	0.67

Table 2: Evaluation of TT extraction on annotated corpora

	Precision	**Recall**	**F1**
GENIA NG terms only (vs. all GENIA terms)			
GNR + MSN	51% (56%)	61% (47%)	0.55 (0.51)
GNR + TF/IDF	45% (51%)	53% (42%)	0.49 (0.46)

Table 3: Comparison to TF/IDF baseline

categories associated. However, the problems of using DBpedia for categorization are

1. to identify the correct domain, e.g. "vitamin C" is related to categories from the biology domain, but also to categories from the music domain
2. to choose an appropriate category if several categories of the same domain are suggested, e.g. "vitamin C" belongs to categories "vitamins", "food antioxidants", "dietary antioxidants", "organic acids", etc.
3. to identify the specificity of the category, e.g. the term "Perineuronal net" is linked to the categories "Neurobiology" and "Neuroscience", where "Neurobiology" also appears as subcategory of "Neuroscience".
4. to categorize instances not found in DBpedia.

To deal with the first two problems, we have evaluated a PMI/IR-based approach, using Turney's [Turney, 2001] 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 input text) and hits retrieved with the choice and the context alone. For evaluating our algorithm, we retrieved the list of DBpedia categories for 100 of our extracted terms with an entry in DBpedia and manually chose a set of no, one or several categories fitting the term in the given context. We then ran our algorithm with three different minimum PMI/IR score thresholds (0, 0.5 and 1) set and compared the output to the manually assigned categories. We then calculated precision, recall and F1 for each of these thresholds and compared the results to two different baselines. Baseline algorithm 1 always assigns the first found DBpedia category, baseline algorithm 2 never assigns any category. The results are summarized in Table 4. Baseline 2 is calculated because only about 22% of the possible categories were assigned by the human annotator. The majority of terms (53%) was not assigned any of the proposed categories. This is because many terms that appeared as entries in DBpedia were not used as technical terms in the given context but in a more general sense. For example, the term "reuse" (appearing in a computer science document), is linked to the categories "waste management" and "technical communication", neither of which fit the given context. Due to this proportion of assigned to non-assigned categories, a PMI/IR threshold of 0 turns out to be too low because it favors assigning a category over not assigning any category. With a threshold of 0, the combined F1 score stays below the baseline score of never assigning any category. With thresholds set to 0.5 and 1, however, the combined F1 score is considerably higher than both baselines. A threshold of 0.5 yields considerably better results for terms with one or more assigned categories and a slightly better overall result than a threshold of 1. The results show that the algorithm can be used to decide whether a proposed DBpedia category fits an instance in the given context or not. In particular, with a PMI/IR score threshold set, it can achieve high precision and recall scores when deciding that a category does not fit a term in the given context.

8 Conclusion and current challenges

We have presented a robust method for domain-independent, unsupervised extraction of TTs from scientific documents with promising results. Up to now, we are not able to categorize all extracted TTs, as is usually done in GN learning, but presented first experimental results towards solving this task. 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. 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 automatically, without using annotated training data. Another major challenge is the categorization of all TTs.

	Thresh = 0	Thresh = 0.5	Thresh = 1	Baseline 1	Baseline 2
Category assignment					
Precision	36.56%	50.00%	48.89%	37.00%	N/A
Recall	53.13%	43.75%	34.38%	57.81%	0.00%
F1	0.43	0.47	0.4	0.45	N/A
No category assignment					
Precision	91.67%	80.43%	71.93%	N/A	53.00%
Recall	20.75%	69.81%	77.36%	0.00%	100.00%
F1	0.34	0.75	0.75	N/A	0.69
Combined results					
Precision	42.86%	63.73%	61.76%	37.00%	53.00%
Recall	38.46%	55.56%	53.85%	31.62%	45.30%
F1	0.41	0.59	0.58	0.34	0.49

Figure 4: Evaluation of DBpedia categorization using different PMI/IR thresholds

Acknowledgments

The research project DILIA (Digital Library Assistant) is co-funded by the European Regional Development Fund (EFRE) under grant number 10140159. We gratefully acknowledge this support.

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*, 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] O. 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.

[Hulth, 2003] A. Hulth. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 216–223, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

[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.

[Mihalcea and Csomai, 2007] R. Mihalcea and A. Csomai. Wikify!: linking documents to encyclopedic knowledge. In *CIKM '07*, New York, NY, USA, 2007. ACM.

[Mihalcea and Tarau, 2004] R. Mihalcea and P. Tarau. TextRank: Bringing order into texts. In *Proceedings of EMNLP-04 and the 2004 Conference on Empirical Methods in Natural Language Processing*, July 2004.

[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*, Orlando, FL, 1999.

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

[Turney, 2000] P. D. Turney. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336, May 2000.

[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*, 2002.