A Comparison of Different Strategies for Automated Semantic Document Annotation

Gregor Große-Bölting
Kiel University, Kiel, Germany
ggb@informatik.uni-kiel.de

Chifumi Nishioka
ZBW – Leibniz Information Centre for Economics
Kiel, Germany
Kiel University, Kiel, Germany
c.nishioka@zbw.eu

Ansgar Scherp
ZBW – Leibniz Information Centre for Economics
Kiel, Germany
Kiel University, Kiel, Germany
a.scherp@zbw.eu

ABSTRACT

We introduce a framework for automated semantic document annotation that is composed of four processes, namely concept extraction, concept activation, annotation selection, and evaluation. The framework is used to implement and compare different annotation strategies motivated by the literature. For concept extraction, we apply entity detection with semantic hierarchical knowledge bases, Tri-gram, RAKE, and LDA. For concept activation, we compare a set of statistical, hierarchy-based, and graph-based methods. For selecting annotations, we compare top-k as well as kNN. In total, we define 43 different strategies including novel combinations like using graph-based activation with kNN. We have evaluated the strategies using three different datasets of varying size from three scientific disciplines (economics, politics, and computer science) that contain 100,000 manually labeled documents in total. We obtain the best results on all three datasets by our novel combination of entity detection with graph-based activation (e. g., HITS and Degree) and kNN. For the economic and political science datasets, the best F-measure is .39 and .28, respectively. For the computer science dataset, the maximum F-measure of .33 can be reached. The experiments are the by far largest on scholarly content annotation, which typically are up to a few hundred documents per dataset only.

CCS Concepts

•Applied computing → Document analysis; Annotation;

Keywords

document annotation; hierarchical knowledge bases

1. INTRODUCTION

Semantic annotation of scientific documents allows for an explicit description of the content's subjects and lays the foundation for advanced analyses and visualizations like topic networks. Traditionally, annotation of scientific documents is conducted by domain experts at libraries or public institutions. For example, in the past the domain experts of Leibniz Information Centre for Economics (ZBW) have manually labeled about 1.6 million documents with on average five annotations per document taken from a large, semantic thesaurus for economics[1] consisting of several thousand descriptors. The goal of manual labeling with semantic concepts is to provide for better search capabilities of the scientific documents on search portals like EconBiz[2]. However, the amount of digitally published documents is tremendously increasing and manual annotation becomes too expensive. Thus, manual document annotation needs to be supported by automatically suggested descriptors.

Past research on automated text annotations used a variety of different natural language processing methods like KeyGraph [17], TextRank [16, 24], and Rapid Automatic Keyword Extraction (RAKE) [19], information retrieval methods such as TF-IDF [22], as well as machine learning approaches like Latent Dirichlet Allocation (LDA) [22] and kNN [10, 21]. However, typically a fairly small number of different strategies is compared in these papers (at most ten strategies). In addition, the existing experiments are typically limited to datasets consisting of a few hundred documents only. In this paper, we address these challenges by introducing a framework for automated semantic document annotation that is composed of four processes for concept extraction, concept activation, annotation selection, and evaluation. The framework is capable of integrating various different methods proposed by the existing literature. Thus, different strategies for automated semantic document annotation can be created and compared.

A total of 43 different strategies are defined using our framework. Some strategies are motivated by the literature and others are new, respectively. Regarding the extraction of concepts from the documents, four methods are integrated in our framework: entity-detection using a semantic knowledge base [10, 15, 12], n-gram [25], RAKE [19], and LDA [22]. Subsequently, we assign scores to the extracted concepts representing its importance for the specific document. We compare two statistical methods for concept activation, namely Frequency [6, 14] and CF-IDF [7]. Beyond them, we also evaluate three hierarchy-based activation methods, namely base activation [12], branch activation [9],

K-CAP 2015, October 07 - 10, 2015, Palisades, NY, USA

ISBN 978-1-4503-3849-3/15/10...$15.00

DOI: http://dx.doi.org/10.1145/2815833.2815838

[1] http://zbw.eu/stw/

[2] http://www.econbiz.eu/

and a novel method called OneHop, all of which exploit the hierarchical structure of a given knowledge base. Finally, we use a set of graph-based methods, the concept's degree [26], HITS [13, 26], and PageRank [16, 18]. Based on the activated concepts, the next process selects the annotations that should be attached to a document from concepts. We compare top-k and the machine learning approach k Nearest Neighbors (kNN). Both methods are capable of continuously integrating new documents and annotations and do not require an expensive training phase. Finally, the quality of the selected annotations are evaluated by comparing it with a gold standard. Here, we aim at answering the research questions: (i) Which strategy overall works best? Which (ii) concept extraction, (iii) concept activation, and (iv) annotation selection method performs best? We apply the standard information retrieval measures, i. e., precision, recall, and F-measure, for evaluating the results.

We have applied the 43 strategies on three different datasets of varying size and origin: The economics dataset (62,924 documents) and political science dataset (28,324 documents) have a gold standard created by experts. The third dataset on computer science (244 documents) uses the authors' and readers' keywords as gold standard. We obtain the best results on all three datasets for the novel strategies combining entity detection with graph-based activation and kNN. To the best of our knowledge, these strategies have not been employed in the literature before. For the economics and political science datasets, the F-measures are .39 and .28, when employing HITS as graph-based activation. For the computer science dataset, the strategy employing entity detection, the graph-based activation method Degree, and kNN results in the F-measure of .33 and slightly outperforms the variant using HITS (F-measure of .32). Regarding the concept extraction methods, the use of entity extraction clearly outperforms the others. Considering the concept activation methods, the graph-based methods work well when kNN is employed and the statistical methods performed well when top-k is applied.

Subsequently, we review related work. Section 3 presents our framework and the employed methods in detail. Section 4 introduces the experiment procedure, the three datasets, and evaluation measures. The results are presented in Section 5 and discussed in Section 6, before we conclude.

2. RELATED WORK

Different fields of research have dealt with automatic document annotation in the past. Natural language processing methods include Rapid Automatic Keyword Extraction (RAKE) [19], KeyGraph [17], and TextRank [16]. They retrieve keywords and keyphrases by analyzing texts and give a score to each of them, in order to generate document annotations. Rose et al. [19] demonstrated that RAKE outperformed TextRank [16] and other state of the art methods. Tuarob et al. [22] experimented with TF-IDF and LDA [4] for annotating observational data records in the field of ecology. They concluded that LDA performed better, if sufficient textual content is available.

Recently, scientific documents in digital libraries are annotated with semantic entities stored in a domain-specific knowledge base [10]. For instance, Medical Subject Headings (MeSH)[3], a knowledge base for the field of medicine, is used as annotations for biomedical articles [10]. A lot of domain-specific knowledge bases like MeSH are hierarchically structured, contain synonyms, and provide rich information about relations among the entities. Thus, existing works have tried to extract entities for annotating documents [10, 12]. Using a knowledge base, there are various methods to give a score to a detected entity. CF-IDF, an extension of the traditional TF-IDF [20], takes into account an importance of an entity in an entire document corpus. Beyond them, methods that exploit a hierarchical graph structure of a knowledge base have been developed [9, 12]. Those methods can reveal concepts that are not directly mentioned in the documents but highly related. In addition, there are graph-based methods that take into account cooccurrence of concepts, concept's degree [26] (number of edges used to connect with other concepts), as well as the well-known web site assessment methods HITS [13, 26] and PageRank [18, 16].

For selecting annotations, various kinds of approaches have been employed such as top-k by Wang et al. [24]. Also machine learning approaches have been employed, e. g., Support Vector Machines (SVM) [2], Naive Bayes, Random Forests [5], Decision Trees [14, 23], and Learning to Rank [10]. Also kNN [10, 21] is used based on the assumption that similar documents share similar annotations. Huang et al. [10] showed that kNN significantly outperforms other machine learning techniques. In their experiments, the authors used a dataset of 1,000 biomedical articles. In addition, kNN does not require an expensive training like the other machine learning approaches [2, 5, 10, 14, 23].

In terms of evaluation datasets, Medelyan [14] evaluated her approaches with 780 publications from agricultural sciences, 500 publications from medicine, and 290 publications from physics. Hulth [11] also conducted the experiments on scientific datasets, containing 2,000 abstracts of articles in the field of computer science with corresponding title and keywords. Trieschnigg et al. [21] performed automated MeSH classifications over 1,000 medical articles using their titles and abstract as input. In this paper, we conduct 43 experiments with three datasets containing in total almost 100,000 documents. That is the by far largest experiments for automated scholarly content annotation.

3. ANNOTATION FRAMEWORK

Our framework for automated document annotation[4] has four processes. First, the framework extracts concepts that represent features of a target document (i .e., concept extraction). Detected concepts get a score (i .e., concept activation). Taking into account computed scores, we select annotations for the document (i .e., annotation selection). Finally, annotations are evaluated by comparing them with a gold standard. Below, we first present the methods implemented in each process. Subsequently, we describe how to create different strategies by configuring the framework.

3.1 Concept Extraction

In this process, concepts that represent features of a target document are detected. Thus, concepts are candidates of annotations. Below, we introduce the four concept extraction methods implemented in our framework:

[3]https://www.nlm.nih.gov/mesh/

[4]Released at: https://github.com/ggb/ShortStories

Entity: Entities (i.e., concepts) are extracted from documents using a domain-specific knowledge base. In a knowledge base, each entity has one or more labels and a unique descriptor, which enables to integrate synonyms into one entity. Based on the labels, entities are extracted from the documents like in [10, 12, 14].

Tri-gram: An n-gram is a contiguous sequence of n words from a document and has long been used in document annotation such as KEA [25]. For the experiments, we set $n = 3$, thus we employ Tri-gram. For instance, there is a document whose preprocessed text is: "paper present empirical analysis". From the text, we extract the phrases {paper, present, empirical, analysis, paper present, present empirical, empirical analysis, paper present empirical, present empirical analysis} as concepts. Thus, we form the union of all unigrams, bi-grams, and tri-grams.

RAKE: Rapid Automatic Keyword Extraction (RAKE) [19] is an unsupervised method for extracting keywords from individual documents. It incorporates word cooccurrence information based on the degree (i. e., connectivity with other words) and frequency of a word. A score of each word is defined as the ratio of degree to frequency. For multi-word phrases, a score is calculated as the sum of the ratio of degree to frequency of each word in the multi-word.

Latent Dirichlet Allocation (LDA): LDA [4] is an unsupervised machine learning technique, which infers latent topics in a document corpus. In the topic model inferred by LDA, each document is represented as a probability distribution over topics, while each topic is again represented as a probability distribution over words. We treat the generated topics as concepts and use them for document annotation like [22]. A score of a concept (i. e., topic) is defined by the probability of the topic. For the experiments, we set the parameters $\alpha = 0.5$, $\beta = 0.1$, the number of topics $k = 100$ along with Griffiths et al. [8] and the number of iterations 100. Following Blei et al. [3], we remove the words that appear in less than 25 different documents in the dataset, to reduce the dimensionality and computational costs.

3.2 Concept Activation

We compute an activation score for each extracted concept by applying three different types of activation methods: statistical methods, hierarchy-based methods, and graph-based methods. Hereafter, we refer to a score of a concept c in a document d by $score(c, d)$.

Statistical Methods. The statistical methods take into account only concepts that are directly mentioned in documents. We introduce the following two methods.

Frequency: As baseline, we introduce Frequency [6, 14], where the $score_{freq}(c, d)$ is simply the number of times c appears in d, for the concept extraction methods Entity and Tri-gram. For RAKE, $score_{freq}(c, d)$ is equal to the original score of a concept c for a document d produced by RAKE (see also above). For LDA, $score_{freq}(c, d)$ is defined by the probability of a topic c for d.

CF-IDF: Compared to the traditional TF-IDF [20] (which is used for document classification by Tuarob et al. [23]), CF-IDF (Concept Frequency Inverse Document Frequency) [7] counts frequencies of concepts instead of term frequencies. As Goossen et al. showed, CF-IDF outperforms TF-IDF on news classification tasks [7]. It is computed as shown below:

$$score_{cfidf}(c, d) = cf(c, d) \cdot log \frac{|D|}{|\{d \in D : c \in d\}|}, \quad (1)$$

where $cf(c, d)$ indicates the ratio of the number of a concept c in d to the number of all concepts in d. D refers to the entire dataset and $|D|$ indicates the number of documents in D. $|\{d \in D : c \in d\}|$ denotes the number of documents that contain c.

Hierarchy-based Methods. The hierarchy-based methods assume that concepts extracted from a document are connected with each other in a hierarchical structure. Exploiting the hierarchical structure, scores of the concepts that are not directly mentioned but connected with an extracted concept are boosted by the following methods.

Base Activation: The base spreading activation [12] boosts scores of concepts in higher levels, based on the assumption that if the document contains a concept (e. g., "apple"), the document is also relevant to its parent concept (e. g., "fruit"). The base activation works as defined in Equation 2.

$$score_{base}(c, d) = freq(c, d) + \lambda \cdot \sum_{c_i \in C_l(c)} score_{base}(c_i, d), \quad (2)$$

where $freq(c, d)$ equals to the number of times c appears in d, and λ and $C_l(c)$ denote the decay parameter and the set of children concepts of a concept c. We set $\lambda = 0.4$ as Kapanipathi et al. suggested [12].

Branch Activation: Different from the base activation, the branch spreading activation [9] takes into account the number of concepts in each level of the hierarchy. Specifically, it normalizes scores by the number of concepts in the higher level as described in Equation 3.

$$score_{branch}(c, d) = freq(c, d) + \lambda \cdot BN \cdot \sum_{c_i \in C_l(c)} score_{branch}(c_i, d), \quad (3)$$

where BN denotes the reciprocal of the number of concepts that are located on one higher level of the level where c is stored. Like Base Activation, we set $\lambda = 0.4$.

OneHop Activation: We introduce the novel OneHop activation method developed in collaboration with domain experts from ZBW. Initially, $score_{onehop}(c, d)$ is given by $freq(c, d)$. In the activation process, if c has two or more children concepts that are directly mentioned in d, c boosted as defined in Equation 4, where C_d denotes a set of concepts that are directly mentioned in a document d. Thus, OneHop activation is limited to activate concepts at maximum one hop distance.

$$score_{onehop}(c, d) = \begin{cases} freq(c, d) + \lambda \cdot \sum_{c_i \in C_l(c)} freq(c_i, d) & \text{if } |C_l(c) \cap C_d| \geq 2 \\ freq(c, d) & \text{otherwise} \end{cases} \quad (4)$$

Graph-based Methods. While the methods presented above are based on a hierarchical structure, we introduce here methods that can be applied to any graph structure. To this end, we use the cooccurrence graph [17], which represents concepts that are close to each other. As an example, we consider the concepts {aleph, argentine writer, short story, poet, jorge l. borges, aleph, short story, aleph, poet} that are extracted in this order from a document. From the extracted concepts, we make pairs (aleph, argentine writer), (argentine writer, short story), (short story, poet), (poet, jorge l. borges), (jorge l. borges, aleph), (aleph, short story), (short story, aleph), and (aleph, poet) along with the order of the concept occurrences. Based on the pairs, the cooccurrence graph is generated as shown in Figure 1. The cooccurrence graph is produced for each document. Subsequently, the concepts are activated using the following methods.

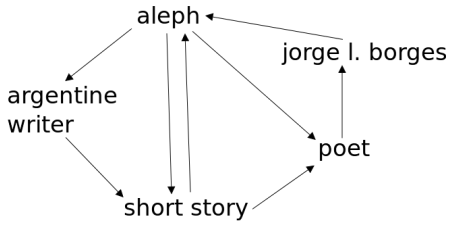

Figure 1: An example of the cooccurrence graph.

Degree: This method takes the degree (i.e., the number of edges) of the concept node as score [27]. Thus, $score_{degree}(c, d) = degree(c, d)$, where $degree(c, d)$ returns the degree of c in the graph for d. For instance, the score of the concept "poet" in Figure 1 is three, because the node has three edges in total.

HITS: The Hyperlink-Induced Topic Search (HITS) [13] is originally introduced as link analysis algorithm for search engines. However, it also showed its effectiveness in keyword extraction for documents [27]. HITS computes two scores called "hub" and "authority" for each node (i. e., concept) in the graph. The idea behind the hub and authority is that a good hub represents a node that point to many other nodes and a good authority is a node linked by many hubs. Initially the two scores are 1. The algorithm traverses the graph and activates scores node by node. The scores "hub" and "authority" are computed by Equation 5 and 6. The final score is computed as the sum of the two scores. Thus, $score_{hits}(c, d) = hub(c, d) + auth(c, d)$.

$$hub(c, d) = \sum_{c_i \in C_n(c)} auth(c_i, d) \quad (5)$$

$$auth(c, d) = \sum_{c_i \in C_n(c)} hub(c_i, d) \quad (6)$$

PageRank: PageRank [18] is another algorithm for search engines and originally measures the importance of a web page. It is based on the intuition that a web page that is linked from many different web pages is more important. PageRank demonstrated its effectiveness for document classification [16]. It is defined below:

$$score_{page}(c, d) = (1 - \mu) + \mu \cdot \sum_{c_i \in C_{in}(c)} \frac{score_{page}(c_i, d)}{|C_{out}(c_i)|}, \quad (7)$$

where μ is a parameter called dumping factor and we set $\mu = 0.85$ as Page et al. [18]. $C_{in}(c)$ and $C_{out}(c)$ denote the set of incoming edges of c and the set of outgoing edges of c, respectively.

3.3 Annotation Selection

This process selects annotations for a document. We evaluate top-k and kNN as discussed in the related work section.

Top-k: Annotation candidates (i.e., concepts) are ranked by scores computed in the previous process. Subsequently, concepts that are ranked above k are selected [24]. We experiment with $k = 5$ and $k = 10$. Since documents are annotated with semantic entities as gold standard (see description of datasets in Section 4.1), the extracted concepts (i.e., keywords) by Tri-gram and RAKE are converted into entities by string matching with entity labels after lemmatization and stopword removal. Thus, only phrases are kept which appear in a knowledge base as entity label.

k Nearest Neighbor (kNN): kNN is a non-parametric algorithm used for classification. It is used for document annotation [10, 21] based on the assumption that documents with similar concept vectors share similar annotations. To conduct kNN, each document is converted into a numerical vector, where each element is a score of a concept. The nearest neighbors are detected by calculating cosine similarities between two documents. A document obtains annotations that are the union of annotations attached to k nearest documents. k is optimized based on the F-measure.

3.4 Configurations

We create different annotation strategies by combining the methods described in the previous sections. However, some methods cannot be combined with certain other methods. The possible configurations and strategies are presented in Table 1. When Entity is employed as concept extraction method, it is possible to apply all methods of each process. Thus, we have 24 strategies (8 concept activation methods $\times$ 3 annotation selection methods) for Entity. Regarding Tri-gram, it is not possible to exploit the hierarchy-based methods, because we do not use any knowledge base for Trigram. This results in 15 strategies (5 concept activation methods $\times$ 3 annotation selection methods). For RAKE and LDA, we cannot use any concept activation methods and use only Frequency. Furthermore, since LDA defines a concept as topic represented by a probability distribution over multiple words, it is not possible to convert a topic to an entity and we cannot apply top-k as annotation selection method. In other words, we only apply kNN when LDA is employed as concept extraction method. Thus, RAKE has three strategies and LDA has one strategy. Therefore, we have 43 strategies in total.

Please note, the framework can be easily extended with further methods at each of the four processes. Thus, it is possible to define new strategies by combining different concept extraction, concept activation, and annotation selection methods.

Table 1: Configuration of strategies. For all strategies, all evaluation measures are applied. In parentheses, we show the number of possible options (e.g., there are two statistical methods).

Concept Extraction	Entity			Tri-gram		RAKE		LDA
Concept Activation	statistical (2)	hierarchy (3)	graph (3)	statistical (2)	graph (3)	Frequency		Frequency
Annotation Selection	top-k (2)	kNN		top-k (2)	kNN	top-k (2)	kNN	kNN

4. EXPERIMENTS

For conducting the experiments using the implemented framework, we preprocessed the documents by lemmatization and stopword removal. Subsequently, we partitioned the dataset into 10 equal sized subsets. Eight subsets are used as training data, one for testing, and one for optimizing the parameter k in the case of kNN. Below, we describe the datasets used in our experiments. Subsequently, we introduce our evaluation measures. In addition, we explain how we apply the evaluation measures in the kNN setting.

4.1 Datasets

We conduct the experiments with three datasets of open access publications that are manually labeled. In addition, we use different knowledge bases for each academic domain.

Economics: The dataset contains 62,924 open access publications in the field of economics. Each publication is annotated on average with 5.26 entities (SD: 1.84) by experts from the ZBW – Leibniz Information Centre for Economics. The annotations come from a domain-specific hierarchical knowledge base, the Standard Thesaurus Economics (STW)[5]. It is a domain-specific knowledge base for economics developed by ZBW. It contains 6,335 entities and 11,679 labels.

Political Science: The dataset is provided by the German Information Network International Relations and Area Studies (FIV)[6] and contains 28,324 publications in the field of political science. On average, a publication has 12.00 entities (SD: 4.02). The corresponding knowledge base is the European Thesaurus on International Relations and Area Studies[7] developed by the European Information Network on International Relations and Area Studies. It contains 7,912 entities and 8,421 labels.

Computer Science: We use the SemEval 2010 Task 5 dataset[8] that contains 244 publications. Originally, 100 publications are for test and 144 publications are for training. In the original dataset, the publications are not annotated with entities but on average 15.00 (string-based) keywords. In order to enable to employ the hierarchy-based methods and to conduct a fair comparison of the strategies over the other datasets, we converted the original keyword-based annotations into entity annotations by string matching with entity labels like Wang et al. [24]. As knowledge base for the field of computer science, we use the ACM Computer Classification System (CCS)[9] in the version from 2012 provided by the Association for Computing Machinery (ACM)[10]. It contains 2,299 entities and 9,086 labels. In result, a publication in the dataset is annotated with 5.05 entities (SD: 2.41).

[5]http://zbw.eu/stw/versions/8.12/about.en.html
[6]http://www.fiv-iblk.de/eindex.htm
[7]https://www.ireon-portal.eu/
[8]http://semeval2.fbk.eu/semeval2.php?location=tasks#T6
[9]http://www.acm.org/about/class/class/2012
[10]https://www.acm.org/

4.2 Evaluation Measures

We evaluate the selected annotations using standard Precision, Recall, and F-measure by comparing it with a gold standard. Precision is defined by the number of correct annotations divided by the total number of annotations made by the strategy. Recall is defined by the number of correct annotations divided by the total number of annotations given by the gold standard. F-measure is the harmonic mean of recall and precision. For the selection method kNN, we first create the union of the annotations of all k nearest neighbors before applying the measures (see Section 3.3).

5. RESULTS

We report the results with respect to each concept extraction method as it is the first process in our framework. We mark the results that show the best performance in bold.

5.1 Entity

Table 2 shows the performance of document annotation with respect to each dataset when Entity is employed as concept extraction method. Regarding concept activation methods, the statistical methods (i.e., frequency and CF-IDF) outperform others when annotations are selected by top-k. However, CF-IDF works much worse than the others for the computer science dataset. When annotations are selected by kNN, the graph-based methods (i.e., OneHop, Degree, HITS, and PageRank) perform best. Especially, the difference between the results of the graph-based methods and others is large in the economics and political science datasets. For the economics and computer science datasets, HITS demonstrates the best performance. In terms of the annotation selection methods, kNN performs better than top-k in the three datasets. Regarding the parameter k for kNN, the optimization resulted in $k = 1$ for the economics and political science datasets and $k = 2$ for the computer science dataset. Entity performs best with the economics dataset regarding F-measure.

5.2 Tri-gram

Table 3 presents the performance of document annotations with respect to each dataset in a same style with Table 2, but when Tri-gram is employed as concept extraction method. In terms of concept activation methods, Frequency performs best in general. However, the concept activation methods almost work equally when kNN is employed. Regarding the annotation selection methods, top-k performs better than kNN for the economics and political science datasets (cf. strategies employing Entity as concept extraction method in Table 2). The optimization for kNN resulted in $k = 2$ for the economics and political science datasets and $k = 3$ for the computer science dataset. Tri-gram performs best with the computer science dataset regarding F-measure.

Table 2: Performance of document annotation with the concept extraction method: Entity.

a) Economics

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	.14 (.17)	.14 (.15)	.13 (.15)	.22 (.20)	.11 (.10)	.14 (.12)	.08 (.21)	.08 (.21)	.08 (.21)
CF-IDF	**.19** (.19)	**.18** (.17)	**.18** (.16)	**.24** (.21)	**.12** (.10)	**.15** (.12)	.29 (.32)	.30 (.32)	.29 (.31)
Base Act.	.10 (.14)	.09 (.13)	.09 (.13)	.18 (.19)	.09 (.09)	.12 (.11)	.20 (.30)	.20 (.30)	.20 (.29)
Branch Act.	.08 (.14)	.08 (.12)	.08 (.12)	.17 (.19)	.08 (.09)	.11 (.11)	.17 (.28)	.17 (.28)	.17 (.27)
OneHop	.12 (.16)	.12 (.14)	.12 (.14)	.19 (.19)	.09 (.09)	.12 (.11)	.35 (.34)	.36 (.34)	.35 (.33)
Degree	.15 (.17)	.14 (.15)	.14 (.15)	.23 (.20)	.11 (.09)	.14 (.12)	.39 (.33)	**.40** (.33)	.38 (.32)
HITS	.14 (.17)	.14 (.15)	.14 (.15)	.23 (.20)	.11 (.10)	.14 (.12)	**.40** (.32)	**.40** (.32)	**.39** (.31)
PageRank	.14 (.17)	.14 (.15)	.14 (.15)	.22 (.20)	.11 (.09)	.14 (.12)	.39 (.33)	**.40** (.33)	.38 (.32)

b) Political Science

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	**.12** (.11)	.18 (.16)	**.14** (.12)	**.15** (.13)	.12 (.10)	**.13** (.10)	.14 (.17)	.05 (.07)	.07 (.09)
CF-IDF	.05 (.07)	.12 (.16)	.07 (.10)	.07 (.09)	.08 (.10)	.07 (.09)	.24 (.22)	.14 (.14)	.17 (.16)
Base Act.	.05 (.08)	.10 (.13)	.07 (.09)	.10 (.10)	.10 (.09)	.09 (.09)	.14 (.19)	.07 (.10)	.09 (.12)
Branch Act.	.04 (.07)	.08 (.12)	.05 (.08)	.09 (.09)	.09 (.09)	.08 (.09)	.12 (.17)	.06 (.10)	.08 (.11)
OneHop	.10 (.09)	**.21** (.17)	.13 (.11)	.13 (.11)	**.14** (.11)	**.13** (.10)	.27 (.21)	.26 (.21)	.25 (.19)
Degree	.10 (.09)	**.21** (.17)	.13 (.11)	.13 (.11)	**.14** (.11)	**.13** (.10)	.29 (.21)	.28 (.21)	.27 (.19)
HITS	.10 (.09)	**.21** (.17)	.13 (.11)	.13 (.11)	**.14** (.11)	**.13** (.10)	**.30** (.22)	**.29** (.21)	**.28** (.20)
PageRank	.10 (.09)	.20 (.17)	.13 (.11)	.13 (.10)	**.14** (.11)	**.13** (.10)	.29 (.22)	**.29** (.21)	.27 (.20)

c) Computer Science

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	**.18** (.21)	**.14** (.15)	**.15** (.16)	**.22** (.22)	**.08** (.08)	**.12** (.11)	.49 (.28)	.24 (.16)	.30 (.17)
CF-IDF	.02 (.08)	.02 (.06)	.02 (.06)	.03 (.11)	.01 (.04)	.02 (.05)	.47 (.29)	.23 (.17)	.29 (.18)
Base Act.	.17 (.20)	.13 (.14)	.14 (.15)	**.22** (.22)	**.08** (.08)	**.12** (.11)	.49 (.28)	.22 (.15)	.29 (.17)
Branch Act.	.17 (.20)	.12 (.14)	.14 (.15)	.21 (.22)	**.08** (.08)	.11 (.11)	**.50** (.28)	.22 (.15)	.29 (.17)
OneHop	.17 (.20)	.13 (.14)	.14 (.15)	.21 (.22)	**.08** (.08)	.11 (.11)	.42 (.30)	.25 (.21)	.29 (.20)
Degree	.17 (.21)	.13 (.15)	.14 (.16)	**.22** (.22)	**.08** (.08)	**.12** (.11)	.49 (.28)	**.27** (.17)	**.33** (.18)
HITS	**.18** (.21)	**.14** (.15)	**.15** (.16)	.21 (.22)	**.08** (.08)	.11 (.11)	.48 (.31)	**.27** (.18)	.32 (.20)
PageRank	.17 (.21)	.13 (.15)	.14 (.16)	**.22** (.22)	**.08** (.08)	**.12** (.11)	**.50** (.29)	.25 (.15)	.31 (.18)

Table 3: Performance of document annotation with the concept extraction method: Tri-gram.

a) Economics

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	**.12** (.15)	**.12** (.14)	**.11** (.14)	**.19** (.19)	**.10** (.10)	**.13** (.12)	.08 (.22)	**.08** (.22)	**.08** (.21)
CF-IDF	.10 (.12)	.10 (.12)	.09 (.11)	.17 (.17)	.08 (.10)	.11 (.12)	.07 (.20)	.06 (.22)	.06 (.20)
Degree	.03 (.09)	.03 (.08)	.03 (.08)	.03 (.09)	.03 (.08)	.03 (.08)	.07 (.21)	.07 (.21)	.07 (.20)
HITS	.02 (.06)	.02 (.06)	.02 (.06)	.02 (.06)	.02 (.06)	.02 (.06)	.08 (.22)	**.08** (.22)	.07 (.21)
PageRank	.03 (.09)	.03 (.08)	.03 (.08)	.03 (.09)	.03 (.08)	.03 (.08)	**.10** (.20)	.04 (.08)	.05 (.11)

b) Political Science

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	**.06** (.08)	**.14** (.16)	**.08** (.10)	**.10** (.10)	**.11** (.11)	**.10** (.09)	.08 (.14)	**.05** (.08)	**.06** (.09)
CF-IDF	.05 (.05)	.06 (.07)	.05 (.06)	.09 (.10)	.09 (.10)	.08 (.09)	.09 (.15)	.04 (.08)	**.06** (.10)
Degree	.01 (.03)	.03 (.07)	.01 (.04)	.01 (.03)	.03 (.07)	.01 (.04)	.11 (.14)	.03 (.05)	.05 (.07)
HITS	.01 (.03)	.02 (.06)	.01 (.03)	.01 (.03)	.00 (.06)	.01 (.03)	**.12** (.14)	.04 (.06)	**.06** (.08)
PageRank	.01 (.04)	.03 (.08)	.02 (.05)	.01 (.04)	.03 (.08)	.02 (.05)	.08 (.12)	.03 (.05)	.04 (.06)

c) Computer Science

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	**.26** (.24)	**.20** (.18)	**.22** (.19)	**.54** (.30)	.20 (.13)	.29 (.17)	.44 (.28)	.25 (.18)	**.30** (.19)
CF-IDF	.23 (.24)	.18 (.18)	.19 (.19)	**.54** (.29)	**.22** (.14)	**.30** (.17)	**.48** (.28)	.20 (.14)	.26 (.15)
Degree	.09 (.15)	.07 (.11)	.07 (.12)	.13 (.19)	.05 (.07)	.07 (.09)	**.48** (.29)	.23 (.16)	.29 (.18)
HITS	.05 (.14)	.04 (.09)	.04 (.10)	.11 (.18)	.04 (.06)	.06 (.09)	.39 (.29)	**.26** (.21)	.28 (.19)
PageRank	.02 (.06)	.02 (.05)	.02 (.06)	.03 (.08)	.01 (.03)	.02 (.05)	.46 (.29)	.25 (.18)	**.30** (.18)

5.3 RAKE

Table 4 describes the performance when RAKE is employed as concept extraction method. As discussed in Section 3.4, we employ only Frequency as concept activation method. In general, we observe that RAKE performs worse than Entity. In terms of the annotation selection method, RAKE demonstrates best results when kNN is employed

Table 4: Performance of document annotation with the concept extraction method: RAKE.

a) Economics

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	.08 (.14)	.08 (.12)	.08 (.12)	.15 (.18)	.07 (.08)	.10 (.11)	.34 (.33)	.34 (.33)	.33 (.32)

b) Political Science

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	.04 (.07)	.08 (.13)	.05 (.08)	.07 (.09)	.08 (.09)	.07 (.08)	.31 (.23)	.18 (.15)	.22 (.17)

c) Computer Science

	top-5			top-10			kNN		
	Recall	**Precision**	**F**	**Recall**	**Precision**	**F**	**Recall**	**Precision**	**F**
Frequency	.24 (.24)	.17 (.16)	.19 (.17)	.42 (.28)	.15 (.10)	.22 (.14)	.42 (.27)	.20 (.13)	.25 (.15)

(like Entity and Tri-gram). The optimization for kNN resulted in $k = 1$ for the three datasets. RAKE demonstrates the best performance with the computer science dataset.

5.4 LDA

Table 5 describes the performance of document annotation based on LDA. As discussed in Section 3.4, we can only apply Frequency as concept activation method and kNN as annotation selection method. We observe that LDA performs in general worse than the other three concept extraction methods. The strategy demonstrates the best performance with the computer science dataset. For the three datasets, the optimization for kNN resulted in $k = 1$.

Table 5: Annotation performance for concept extraction with LDA and annotation selection kNN.

a) Economics

	kNN		
	Recall	**Precision**	**F**
Frequency	.19 (.30)	.19 (.30)	.19 (.30)

b) Political Science

	kNN		
	Recall	**Precision**	**F**
Frequency	.15 (.19)	.15 (.18)	.14 (.17)

c) Computer Science

	kNN		
	Recall	**Precision**	**F**
Frequency	.28 (.27)	.24 (.23)	.24 (.22)

6. DISCUSSION

We discuss the results shown in the previous section along the research questions outlined in Section 1.

Best Performing Strategy. Looking into the F-measure, the strategy which employs Entity as concept extraction method, HITS as concept activation method, and kNN as annotation selection method performs best for the economic and political science datasets with the F-measures of .39 and .28, respectively. For the computer science dataset, the strategy which employs Entity, Degree as concept activation method, and kNN performs best with the F-measure of .33. Thus, we can state that in general our novel combination of graph-based activation methods (e. g., HITS and Degree) in combination with entity detection and kNN outperforms the other strategies.

Influence of Concept Extraction Methods. Regarding the concept extraction methods, i. e., Entity, Tri-gram, RAKE, and LDA, we observe that Entity consistently outperforms the others on all datasets. Thus, concepts defined in a domain-specific knowledge base can assist document annotation [10, 12, 15]. An explanation is that the concepts avoid detecting noisy entities that are not relevant to the documents [1]. Such knowledge-bases are available for free and in high quality for many domains.[11] They are highly adopted in library sciences and related communities.

Influence of Concept Activation. We applied different concept activation methods when employing Entity and Tri-gram as concept extraction method. While previous works like Kapanipathi et al. [12] showed that hierarchy-based methods performed well, our experiments document that hierarchy-based methods generally performed lower than graph-based methods (see Table 2). A possible reason is that different from the previous work [12], we used the full texts of scientific publications in the experiments. Full texts already contain so many different concepts that other concepts that are not directly mentioned do not have to be activated. For instance, we detected on average 203.80 unique entities (SD: 24.50) employing Entity as concept extraction method in the economics dataset. Therefore, the statistical methods worked better than the hierarchy-based methods. While the statistical methods perform well when employing top-k as annotation selection method, the graph-based methods even work better when applying kNN as annotation selection method. Graph-based methods require computing the cooccurence graph over the entities. Thus, finally it is noteworthy that the much simpler OneHop activation introduced in this paper already achieves very similar results in terms of F-measure but at much lower computational costs.

Influence of Annotation Selection. In our experiments, we compared the three annotation selection methods: top-5, top-10, and kNN. We can state that in general kNN enhances the performance (except one strategy using Frequency, where the results are overall very low). We believe that our results can be generalized to other domains that are of similar characteristics like social science, information science, and psychology. In particular if gold standard annotations are of similar style and a thesaurus is available. We base this assumption on (a) the large-scale datasets of schol-

[11] A list is maintained at W3C: http://www.w3.org/2001/sw/wiki/SKOS/Datasets

arly content used in this experiments (almost two orders of magnitude larger than any results documented so far). This makes the results robust and stable. In addition, (b) results obtained in a field very different from our domains, namely on 2,000 articles in biomedicine [10] and 1,000 articles in medicine [21] also determine kNN as the best performing selection method. However, both studies did not employ graph-based activation methods like in this paper that have shown even higher improvements for kNN.

7. CONCLUSIONS

We have performed extensive empirical analyses of 43 different strategies for automated document annotation. We used three datasets of different size and scientific offspring (economics, politics, and computer science) with a total of almost 100,000 manually labeled documents. We found out that on all three datasets a novel combination of graph-based concept activation methods and kNN as concept selection method made the best document annotations in terms of F-measure. The overall best results are obtained for the strategy combining Entity with HITS as graph-based concept activation and kNN (F-measure for economic dataset: .39, political science dataset: .28). For the computer science dataset, the strategy employing Entity with Degree as graph-based concept activation method and kNN performs slightly better (F-measure: .33) than using HITS (F-measure: .32).

Acknowledgments. We thank Karin Wortmann and Tobias Rebholz from ZBW for invaluable discussions and feedback and the economics dataset. We also thank Petra Galle and Robert Strötgen from the German Institute for International and Security Affairs and Jan Lüth from the German Institute of Global Area Studies for the political science dataset. Finally, we thank Henrik Schmidt for his support in analyzing the economics dataset.

8. REFERENCES

[1] F. Abel, E. Herder, and D. Krause. Extraction of professional interests from social web profiles. In *Augmenting User Models with Real World Experiences to Enhance Personalization and Adaptation*, pages 1–6. Springer, 2011.

[2] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[3] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *ICML*, pages 113–120. ACM, 2006.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[5] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *COLT*. ACM, 1998.

[6] A. Csomai and R. Mihalcea. Linguistically motivated features for enhanced back-of-the-book indexing. In *ACL*, pages 932–940, 2008.

[7] F. Goossen, W. IJntema, F. Frasincar, F. Hogenboom, and U. Kaymak. News personalization using the CF--IDF semantic recommender. In *WIMS*. ACM, 2011.

[8] T. L. Griffiths and M. Steyvers. Finding scientific topics. *PNAS*, 101(suppl 1):5228–5235, 2004.

[9] G. Grosse-Bölting, C. Nishioka, and A. Scherp. Generic process for extracting user profiles from social media using hierarchical knowledge bases. In *ICSC*, pages 197–200. IEEE, 2015.

[10] M. Huang, A. Névéol, and Z. Lu. Recommending MeSH terms for annotating biomedical articles. *JAMIA*, 18(5):660–667, 2011.

[11] A. Hulth. *Combining machine learning and natural language processing for automatic keyword extraction*. PhD thesis, Stockholm University, 2004.

[12] P. Kapanipathi, P. Jain, C. Venkataramani, and A. Sheth. User interests identification on Twitter using a hierarchical knowledge base. In *ESWC*, pages 99–113. Springer, 2014.

[13] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[14] O. Medelyan. *Human-competitive automatic topic indexing*. PhD thesis, The University of Waikato, 2009.

[15] O. Medelyan, E. Frank, and I. H. Witten. Human-competitive tagging using automatic keyphrase extraction. In *EMNLP*, pages 1318–1327. Association for Computational Linguistics, 2009.

[16] R. Mihalcea and P. Tarau. TextRank: Bringing order into texts. In *EMNLP*. Association for Computational Linguistics, 2004.

[17] Y. Ohsawa, N. E. Benson, and M. Yachida. KeyGraph: Automatic indexing by co-occurrence graph based on building construction metaphor. In *ADL*, pages 12–18. IEEE, 1998.

[18] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[19] S. Rose, D. Engel, N. Cramer, and W. Cowley. Automatic keyword extraction from individual documents. In *Text Mining: Theory and Applications*, pages 1–20. John Wiley and Sons, 2010.

[20] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[21] D. Trieschnigg, P. Pezik, V. Lee, F. De Jong, W. Kraaij, and D. Rebholz-Schuhmann. MeSH Up: effective MeSH text classification for improved document retrieval. *Bioinformatics*, 25(11), 2009.

[22] S. Tuarob, L. C. Pouchard, and C. L. Giles. Automatic tag recommendation for metadata annotation using probabilistic topic modeling. In *JCDL*, pages 239–248. ACM, 2013.

[23] S. Tuarob, C. S. Tucker, M. Salathe, and N. Ram. Discovering health-related knowledge in social media using ensembles of heterogeneous features. In *CIKM*, pages 1685–1690. ACM, 2013.

[24] R. Wang, W. Liu, and C. McDonald. How preprocessing affects unsupervised keyphrase extraction. In *CICLing*, pages 163–176. Springer, 2014.

[25] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning. Kea: Practical automatic keyphrase extraction. In *DL*. ACM, 1999.

[26] A. Zouaq, D. Gasevic, and M. Hatala. Voting theory for concept detection. In *ESWC*, pages 315–329. Springer, 2012.

[27] A. Zouaq, D. Gasevic, and M. Hatala. Voting theory for concept detection. In *ESWC*. Springer, 2012.