Document Analysis and Retrieval Tasks in Scientific Digital Libraries

Sujatha Das Gollapalli*, Cornelia Caragea†, Xiaoli Li*, C. Lee Giles‡

*Institute for Infocomm Research,
Agency for Science and Technology Research, Singapore
{gollapallis,xlli}@i2r.a-star.edu.sg
†Computer Science and Engineering,
University of North Texas, U.S.A.
ccaragea@unt.edu
‡Information Sciences and Technology,
Computer Science and Engineering,
The Pennsylvania State University, U.S.A.
giles@ist.psu.edu

Abstract. Machine Learning (ML) algorithms have opened up new possibilities for the acquisition and processing of documents in Information Retrieval (IR) systems. Indeed, it is now possible to automate several labor-intensive tasks related to documents such as categorization and entity extraction. Consequently, the application of machine learning techniques for various large-scale IR tasks has gathered significant research interest in both the ML and IR communities. This tutorial provides a reference summary of our research in applying machine learning techniques to diverse tasks in Digital Libraries (DL). Digital library portals are specialized IR systems that work on collections of documents related to particular domains. We focus on open-access, scientific digital libraries such as CiteSeer^x, which involve several crawling, ranking, content analysis, and metadata extraction tasks. We elaborate on the challenges involved in these tasks and highlight how machine learning methods can successfully address these challenges.

Keywords: classification, focused crawling, PageRank, citations, topic modeling, information extraction

1 Introduction

Digital libraries are IR systems that work on collections of documents related to specific domains and involve several information retrieval, information extraction (IE) and graph analysis tasks. While the IR tasks in digital libraries pertain to identifying relevant documents to be indexed and facilitating search functionalities, the IE tasks address the extraction of structured data and entities from unstructured documents. The extracted entities are further used for diverse data mining applications such as link analysis, community detection, and author profiling.

Scientific digital library portals such as CiteSeer^x [39] and ArnetMiner [55] are large-scale IR systems that work on collections of research literature related to specific disciplines. These systems are non-commercial, **open-access** systems that employ automated techniques to identify and index freely-available Web documents related to scientific research and researchers. The metadata extracted from these document collections is used to construct author-document, citation and co-authorship graphs that facilitate different search applications. Moreover, scientometric and bibliometric measures are estimated based on these extracted networks [1]. Needless to say, the satisfaction of users of such DLs and the accuracy of the extracted networks and computed measures depend *crucially* on acquisition and error-free "parsing" of the relevant documents and webpages.

- What URLs need to be examined on the Web to obtain the research-related content for indexing?
- Given a webpage containing links to research-related and other content, how can we filter out the "noise"?
- Given a crawled document, can we accurately determine if it is a research article and, if so, automatically extract the titles, authors, and its keywords?
- Can we generate author profiles based on the documents associated with a researcher?
- How can we disambiguate researchers having the same name while building citation and co-authorship networks?
- What kind of communities, temporal and topical trends do document and co-authorship networks exhibit?

In this tutorial, we draw on our research related to various modules in CiteSeer^x[1] and show that ML techniques can be employed to answer several of the above questions with reasonable accuracy. CiteSeer^x is an open-source digital library portal for scientific and academic papers for Computer Science and related areas. CiteSeer^x is widely considered the first search engine for academic paper search and a predecessor to Google Scholar[2] and Microsoft Academic Search[3]. With the objective of rapid dissemination of scientific scholarly knowledge, CiteSeer^x currently indexes over a million scholarly documents and provides various functionalities such as automatic citation indexing, author disambiguation, reference linking, and metadata extraction over these documents.

We consider the tasks - **classification**, **metadata extraction**, and **content analysis** in this tutorial. Over the last two decades, these topics have received considerable interest in the ML community due to their applicability in diverse domains including the Web, Biology, Politics, and Law [41, 40, 31, 13]. Consequently, efficient, state-of-the-art ML algorithms are now available for solving these tasks. Applying ML algorithms within scientific digital libraries pose significant challenges. Several issues pertaining to scalability, feature design, noise

[1] http://citeseerx.ist.psu.edu/
[2] http://scholar.google.com/
[3] http://academic.research.microsoft.com/

and multiple modalities in the input data need to be dealt with for applying ML algorithms for digital library tasks.

Organization: In each section of this tutorial, we present an overview and challenges related to one of the tasks–classification, metadata extraction, and content analysis in the context of digital libraries. We describe the commonly-used ML techniques for these tasks in CiteSeer^x and other comparable systems. This tutorial provides a reference summary of the material presented at the 8th Russian Summer School in Information Retrieval (RuSSIR 2014)[4].

Section 2 describes classification models for identifying researcher homepages and scientific documents in Computer Science. In Section 3, we describe the extraction of researcher information from homepages and the extraction of keyphrases from research papers. Finally, in Section 4, we present a summary on the usage of topic modeling tools for content analysis and ranking tasks in digital libraries. We focus on discussing semi-supervised and unsupervised ML techniques with the objective of reducing requirements for human-labeled data for learning accurate models.

2 Identifying Research Documents

Classification modules comprise core components in digital libraries. Given that digital library portals provide domain-specific search functionalities on specialized document collections, *how can we ensure that only relevant documents are indexed in the digital libary collection?* Publishers such as ACM DL[5] and PubMed[6], depend on manually-provided information and "trusted" sources to obtain and maintain documents relevant to their respective domains. In contrast, Web crawlers are employed for obtaining publicly-available documents from the Web that are relevant to the domain in open-access systems such as CiteSeer^x.

A Web crawler is a special software that systematically pulls content from the World Wide Web for the purpose of indexing it locally [6]. Given the infeasibility of examining the entire content on the ever-changing Web, *how do open-access IR systems ensure that their indexed collections are relevant and up-to-date?* Periodic and focused crawling of websites where relevant documents are likely to be found is employed for this purpose. A focused crawler aims at minimizing the use of network bandwidth and hardware by selectively crawling only pages relevant to a (specified) set of topics. A key component in such a crawler is a classification module that identifies whether a webpage being accessed during the crawl process is potentially useful to the collection [7].

For a digital library portal such as CiteSeer^x, the size and quality of the indexed collection depends on the accurate identification of various **research-related documents** during periodic crawls of "whitelist academic URLs" [39, 56, 58]. The relevant documents for CiteSeer^x include scientific publications and researcher-related webpages such as professional homepages. Given the URL

[4] http://romip.ru/russir2014/
[5] http://dl.acm.org/
[6] http://www.ncbi.nlm.nih.gov/pubmed

of a website where such documents are typically hosted, *how can we automatically identify researcher homepages from "irrelevant" pages such as course pages, seminar postings, and other academic webpages. Similarly, given a crawled document, can we automatically identify whether it is a research article or non-research article?* We present some studies on CiteSeer^x that address these questions using novel, problem-specific features.

The identification of researcher homepages in the context of the ever-changing Web was addressed in [17]. In this work, the authors raised the concern of training classifiers when labeled datasets do not provide sufficient coverage of "negative" or "irrelevant" documents. For example, although WebKB[7] is a well-known labeled dataset used for training researcher homepage classifiers, due to its outdated nature[8], content-based classifiers trained on WebKB were found to be inaccurate for classifying content on the current-day websites. This low-performance was attributed to the presence of new types of webpages corresponding to jobs, code, seminars, calendars, and lecture material available on current-day academic websites [49]. The newer types of webpages are different from the types covered in the WebKB labeled dataset, which contains faculty and student homepages, course, staff, and project pages [17].

Various features based on surface-patterns and term presence in the WordNet[9] dictionaries were designed from the URL strings and shown to be more consistent across datasets for discriminating non-homepages [17]. This aspect is illustrated in Table 1 from [17]. The table highlights the overlap in the top URL and content features based on Information Gain between the training and crawled datasets. The low overlap in content features in the two columns illustrates why these features are not discriminative in identifying researcher homepages in the newer crawls when a classifier trained on WebKB is used.

Table 1. Overlap in the top URL and content features based on Information Gain between training and crawl datasets

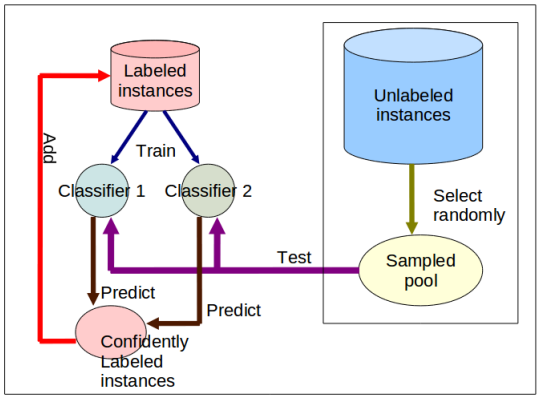
URL		Content	
training	crawl	training	crawl
TILDENODICT	**ALPHANUMBER**	gmt	**university**
TILDENODICT_SEQEND	**TILDENODICT**	server	computer
ALPHANUMBER	ALPHANUMBER_ALPHANUMBER	type	science
NONDICTWORD	HYPHENATEDWORD	html	department
courses	**ALPHANUMBER_SEQEND**	content	numImages
ALPHANUMBER_SEQEND	**TILDENODICT_SEQEND**	text	numLinks
users_NONDICTWORD	QMARK	date	cs
users	NUMBER	professor	box
NONDICTWORD_SEQEND	courses	**university**	ri
homes	NUMBER_SEQEND	research	providence

In [17], techniques for improving content-based classification of researcher homepages by bootstrapping with URL features are discussed. Multiview learn-

[7] http://www.cs.cmu.edu/afs/cs/project/theo20/www/data/
[8] The WebKB dataset was created in 1997.
[9] http://wordnet.princeton.edu/

ing and particularly **co-training** that works with independent views of the learning instances is used for researcher homepage classification by treating URL and content-based features as independent views of the data. The algorithm is shown via a schematic diagram in Figure 1. The co-training procedure starts by train-

Fig. 1. Illustration of co-training. Credit: http://web.cs.gc.cuny.edu/~zhengchen/papers/naacl09-bootstrap-slides.ppt



ing two classifiers on independent feature views of a small number of labeled classification instances. In every subsequent iteration, predictions made on unlabeled instances by the two classifiers are used to expand the training dataset for the next round of training. Co-training was shown to significantly improve content-based researcher homepage classification when compared to other semi-supervised algorithms that use a single view of webpages (URL+content features together) (see [17] for more details).

Although text and bag-of-words (BoW) representations are common in text classification problems [46, 25], as illustrated in the researcher homepage identification problem in CiteSeer^x, it is often beneficial in digital libraries to design features based on the particular problem and domain rather than an "off-the-shelf" application of existing techniques. We describe another classification task in CiteSeer^x where improvements beyond BoW are obtained using simple structural features specific to research documents.

In CiteSeer^x, it is desirable to accurately identify research articles from a set of crawled documents and index these articles in the library for fast search and retrieval of information. A rule-based system that classifies documents as research articles if they contain any of the words `references` or `bibliography` will mistakenly classify documents such as curriculum vita or slides as research articles whenever they contain the word `references` in them, and will fail to identify research articles that do not contain any of the two words. On the other hand, the commonly used "bag of words" representation for document

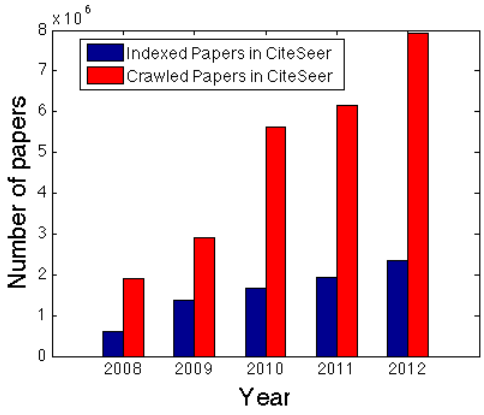


Fig. 2. The growth in the number of crawled documents as well as in the number of research papers indexed by CiteSeer$^{\mathrm{x}}$ between 2008 and 2012.

classification can result in prohibitively high-dimensional input spaces. Machine learning algorithms applied to these input spaces may be intractable due to the large number of dimensions. In addition, the "bag of words" may not capture the specifics of research articles, e.g., due to the diversity of the topics covered in CiteSeer$^{\mathrm{x}}$. As an example, an article in Human Computer Interaction may have a different vocabulary space compared to a paper in Information Retrieval, but some essential terms may persist across the papers, e.g., "references" or "abstract". The number of tokens in a document could be also very informative, i.e., the number of tokens in a research article is generally much higher than in a set of slides, but much smaller than in a PhD thesis. However, these aspects are not captured by the "bag of words" representation.

The number of crawled documents in CiteSeer^{x} are in the order of millions. Figure 2 shows the increase in both the number of crawled documents as well as the number of research articles indexed by CiteSeer$^{\mathrm{x}}$ between 2008 and 2012. As can be seen from the figure, the number of crawled documents has increased from less than two million to almost eight million, whereas the number of indexed documents has increased from less than one million to more than two million. Due to this scale and the problems described in the previous paragraph, "bag of words" approaches may not be efficient for run-time handling of research article identification in CiteSeer^{x}. To handle these challenges, novel features, called structural features, extracted from the content and the structure of crawled documents were proposed in [5]. Some of these features include keywords such as "abstract", "references", "bibliography", "introduction", $n-$gram features such as "this paper", "this report", "this manual", and other features such as "number of pages", "number of words in the document", "percentage of space" and "number of lines per page".

3 Metadata Extraction

Digital libraries often work with multiple types of documents. For instance, in CiteSeer^x, both research publications and technical reports (usually in PDF format) are crawled along with researcher homepages (HTML). Once again, in contrast with systems such as ACM DL and PubMed that work on human-provided clean metadata, various supervised, semi-supervised and unsupervised techniques are employed in automated systems to extract metadata and other information from pdfs and webpages. For example, classification models trained using Support Vector Machines [4] are used to extract the title and author information from the header of a research publication [24]. Similarly, sequential modeling is employed using Conditional Random Fields (CRFs) [38] in the ParsCit tool for extracting citations and the structure from a scientific document [9]. CRFs are also employed to extract researcher metadata such as email, university affiliations, and job positions from their homepages in ArnetMiner [55].

Although supervised learning models that are trained on human-annotated data are popular for learning metadata parsers, sometimes, simple rules and heuristics can be employed for the same. For instance, based on the heuristic that the first 'person' name in a researcher homepage corresponds to the researcher, the Named Entity Recognition tool from Stanford[10] was used directly to extract researcher names from homepages [18]. Similarly, regex patterns are effective for extracting phone and fax numbers from webpages [53, 55]. In general, accurate rules are desirable in digital libraries since supervised learning techniques require large amounts of labeled training data that are tedious to annotate for information/metadata extraction tasks [53]. In this section, we describe some weakly-supervised and unsupervised techniques that can be used to extract certain types of information from scientific documents.

Consider the task of extracting metadata fields: *employment position*, *university*, *department affiliations* and *contact information* such as email, phone and fax from a researcher homepage. The corresponding sequence labeling problem (also known as tagging problem or annotation problem) involves predicting for each token from the content of a homepage, a tag/label from the set: {AFFL, EMAIL, FAX, PHN, POS, UNIV, O} where these labels correspond to "affiliation", "email id", "fax number", "phone number", "employment position", "university" and "other" fields, respectively. An example is illustrated in Table 2.

Previous research on this task showed that tagging or sequence labeling approaches out-perform classification approaches due to dependencies among the tags [55, 60]. For instance, it is common to find employment position information followed by the affiliation information on a homepage (e.g., "professor" in the "Computer Science department" at "Stanford"). Such dependencies are captured via sequential models rather than classification techniques that make predictions for a given token position independent of the predictions for neighboring tokens.

Consider sample cue words show in Table 3 that typically surround researcher metadata on their homepages. *Can these cue words provide "weak supervision"*

[10] http://nlp.stanford.edu/ner/index.shtml

Table 2. Example illustrating homepage tagging.

I	**am**	**a**	**student**	**at**	**Penn**	**State**	**and**	**work**
O	O	O	POS	O	UNIV	UNIV	O	O
with	**Professor**	**Xxxxx**	**Yyyyy**	**on**	**finite**	**state**	**automata**	...
O	O	O	O	O	O	O	O	...

while learning annotation models without having to train on fully-annotated examples? Mann, Druck and McCallum [12] proposed **feature labeling** to answer this question. Rather than fully-annotated instances, "weak supervision" provided via (feature, label) affinities were employed by them to train discriminative classification and tagging models [44, 12].

Consider the example in Table 2. Even without annotating the entire snippet, from domain knowledge, one can expect the correct label for the token "student" to be "POS", "most" of the time. This hint can be imposed as a soft preference or a constraint by specifying the (feature, label) distribution. For example, the labeled feature "student POS:0.8, O:0.2", indicates a preference for marking the token "student" with the label "POS" 80% of the time. The probability for the "O" tag is to capture scenarios when the token does not indicate a position on the webpage. For example, the homepage could belong to a researcher who mentions a list of his current "students" as opposed to a student's homepage where the position information is indicated as "graduate student".

Table 3. Sample cue words for different metadata fields.

AFFL: center, centre, college, department, dept, dipartimento, laboratory
UNIV: universiteit, universitat, university, univ
PHN: cell, ext, extn, homephone, mobile, numbers, ph, phonefax, phone
FAX: ext, extn, facsimile, fax, faxno, faxnumber, telefax, tel/fax
EMAIL: contact, email, firstname, lastname, gmail, mail, mailbox, mailto
POS: president, prof, professor, gradstudent, researcher, scholar, scientist,

Generalized Expectation (GE) and Posterior Regularization (PR) are two frameworks studied previously for imposing preferences expressed as labeled features while learning discriminative models [45, 15]. Using the same amount of labeled data, researcher metadata extraction accuracy was improved by $2-8\%$ by adding weak supervision via various term and layout-specific labeled features in [21]. Labeled features effectively reduce training data requirements while learning researcher metadata extraction from their homepages.

In addition to the extraction of structured metadata such as researcher information, other types of information is desirable from research papers in digital libraries. For example, the "concepts" in such papers are not always provided directly with these papers. However, accurate extraction of such concepts (or keyphrases) from research papers can allow for *efficient* processing of more infor-

mation in less time for top-level data mining applications on research document collections such as topic tracking, information filtering, and search.

Keyphrase extraction is defined as the problem of automatically extracting descriptive phrases or concepts from a document. *Keyphrases* act as a concise summary of a document and have been successfully used in several data mining, machine learning and information retrieval applications such as query formulation, document clustering, recommendation, and summarization [32, 59, 23, 51].

Keyphrase extraction was previously studied using both supervised and unsupervised techniques for different types of documents including scientific abstracts, newswire documents, meeting transcripts, and webpages [14, 30, 48, 42, 47]. Based on recent experiments in [37, 36, 3], the PageRank family of methods and *tf-idf* based scoring can be considered the state-of-the-art for unsupervised keyphrase extraction. We describe CiteTextRank, a fully unsupervised graph-based algorithm that incorporates evidence from multiple sources (citation contexts as well as document content) in a flexible manner to score keywords for keyphrase extraction in CiteSeer^x [16].

Let T represent the types of available contexts for a document, d. These types include the *global* context of d, $\mathcal{N}_d^{Ctd}$, the set of *cited* contexts for d, and $\mathcal{N}_d^{Ctg}$, the set of *citing* contexts for d. The global context of d refers to the document's content whereas cited and citing contexts refer to the short text segments around citations to the document d and made by d in the overall document network. An undirected graph, $G = (V, E)$ for d is constructed as follows:

1. For each unique candidate word extracted from all available contexts of d, add a vertex in G.
2. Add an undirected edge between two vertices v_i and v_j if the words corresponding to these vertices occur within a window of w contiguous tokens in any of the contexts.
3. The weight w_{ij} of an edge $(v_i, v_j) \in E$ is given as

$$w_{ij} = w_{ji} = \sum_{t \in T} \sum_{c \in C_t} \lambda_t \cdot \text{cossim}(c, d) \cdot \#_c(v_i, v_j) \tag{1}$$

where $\text{cossim}(c, d)$ is the cosine similarity between the *tf-idf* vectors of any context c of d and d [46]; $\#_c(v_i, v_j)$ is the co-occurrence frequency of words corresponding to v_i and v_j in context c; C_t is the set of contexts of type $t \in T$; and λ_t is the weight for contexts of type t.

The vertices in G (and the corresponding candidate words) are scored using the PageRank algorithm [50]. That is, the score s for vertex v_i is obtained by recursively computing the equation:

$$s(v_i) = (1 - \alpha) + \alpha \sum_{v_j \in Adj(v_i)} \frac{w_{ji}}{\sum_{v_k \in Adj(v_j)} w_{jk}} s(v_j) \tag{2}$$

where α is the damping factor typically set to 0.85 [26].

Unlike simple graph edges with fixed weights, notice that the above equations correspond to *parameterized* edge weights. The notion of "importance" of

contexts of a certain type is incorporated using the λ_t parameters. For instance, one might assign higher importance to citation contexts over global contexts, or cited over citing contexts. One way to visualize the edges is to imagine the two vertices in the underlying graph to be connected using multiple edges of different types. For example, in Figure 3, the two edges between "logic" and "programming" could correspond to *cited* and *global* contexts respectively.

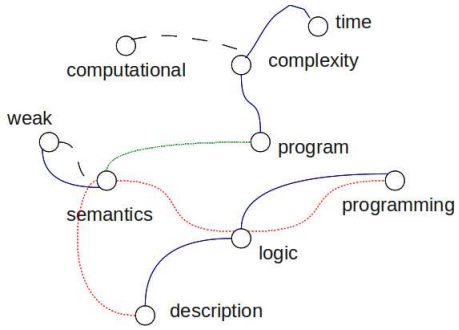


Fig. 3. A small word graph shown in [16]. The edges added due to different context types are shown using different colors/line-styles

We refer the reader to [16] for more details. It was shown in this work that including information from the interlinked document network available in CiteSeer^x provides statistically significant improvements over the existing state-of-the-art models for keyphrase extraction [16].

4 Content Analysis using Topic Models

So far, we discussed "document-level" identification and extraction tasks in digital libraries. *How can we obtain a "macro view" of a given document collection without analyzing each document?* Clustering along with visualization and ontology extraction techniques are often employed in IR systems for obtaining aggregate views of the underlying document collections [46]. Most clustering techniques represent documents using bag-of-words techniques. For example, vector-space models use vectors in high-dimensional term spaces to represent documents [52].

In contrast to the vector view, probabilistic modeling expresses each document using multinomial distributions on terms where each document is assumed to belong to one of the latent topics in the collection [43, 6]. More recently, however, documents are being modeled as "topical mixtures" where a document can potentially cover multiple topics. Given a document collection, the latent concepts or "topics" can be extracted by applying techniques from Linear Algebra on the underlying term-document matrices [10, 46].

Unsupervised models such as Latent Semantic Indexing (LSI) and their probabilistic counterparts such as Latent Dirichlet Allocation (LDA) and probabilistic Latent Semantic Indexing (pLSI) extract latent topics or concepts in a document collection and estimate probability distributions on terms in the vocabulary for each topic [29, 2]. In these models, each document can be associated with a vector in a low-dimension space corresponding to the topics in the collection. Previous studies show the effectiveness of LDA and LSI models in analyzing text corpora in terms of its topics for a multitude of applications (for example, [8, 35, 33, 57]). We discuss the application of topic modeling including LDA and its extensions in a few tasks related to digital libraries.

We refer the reader to [2, 22, 28] for details on the document generation process and parameter estimation in LDA. Here, we describe the output from an LDA run for gaining the intuition behind topic models. Given a collection of documents and the number of topics, as part of parameter estimation, LDA outputs a topic-term association matrix, $\boldsymbol{\phi}$, of size $K \times V$, where K is the number of topics and V, the size of the vocabulary. The entries of $\boldsymbol{\phi}$ correspond to predictive distributions of words/terms given topics. That is, $\phi_{w,i}$ is the probability of a word w given the topic i. These probabilities can be used to express a document, d as a mixture of topics, $\boldsymbol{\theta}_d$. This K-component vector captures the proportion of each topic in the given document. The terms with high probabilities for a given topic in ϕ, upon manual examination, often indicate the underlying concept captured by a topic in the given corpus.

LDA was used to understand the content of an average Computer Science researcher homepage in CiteSeer^x. Table 4 shows the top words of topics indicative of homepages obtained with LDA on a dataset of researcher homepages obtained from DBLP [18]. Notice that the terms in these topics capture information related to contact information, teaching and professional activities of a researcher. In addition, as illustrated in the topics shown in Table 5 obtained in the same run of LDA, it seems typical for Computer Science researchers to mention information related to their research projects and publications on their homepages.

Table 4. Top words of topics related to homepages

talk	page	students	member
slides	home	graduate	program
invited	publications	faculty	committee
part	links	research	chair
talks	contact	cse	teaching
tutorial	personal	student	board
seminar	list	undergraduate	editor
summer	updated	college	courses
book	fax	current	state
introduction	email	ph	activities

The topic-term probabilities estimated using LDA were used to identify researcher homepages among other types of webpages [18]. In addition, the topics corresponding to subject areas (Figure 5) were used to rank fixed-length text segments in a homepage to extract text segments corresponding to research descriptions. Let t be a topic related to a subject area and w, a word inside a text segment, s. The score for s with respect to a topic t is given by

$$\texttt{score}(s,t) = \sum_{w \in s} \phi_{w,t}$$

The research description segment is extracted using

$$p = \underset{t \in ST, s \in S}{argmax} \texttt{score}(s,t)$$

where S is all possible segments in the homepage with a given size sz and ST is the set of all topics indicating subject areas. Anecdotal examples of research descriptions extracted using this method from [18] are shown in Figure 4.

4.1 Improving ranking tasks using topic models

The insights from the topics extracted by LDA models can be used to improve diverse ranking and recommendation tasks in digital libraries. For example, the terms identified for homepage topics were combined effectively with other features based on URLs and HTML structure to train a ranking function for ranking homepages in response to researcher name queries [19]. Citation links were incorporated into extended LDA models for identifying author interests and influence [35], citation recommendation [34] and for identifying topical trends over time [27].

More commonly, authors in digital libraries can be represented in terms of their term distributions or topical profiles obtained with LDA [22, 54, 11]. We describe a graph-based model for scoring authors for expert ranking and similar expert search using the output from LDA [20]. Let T be the set of all topics for a given collection of documents. Intuitively, an expert on a topic, $t \in T$ would

Table 5. Top words from topics related to subject areas

data	multimedia	systems	design
database	content	distributed	circuits
databases	presentation	computing	systems
information	document	peer	digital
management	media	operating	signal
query	data	grid	vlsi
systems	documents	storage	ieee
xml	based	middleware	hardware
acm	hypermedia	system	fpga
vldb	video	scale	implementation

Fig. 4. Sample research description segments extracted from homepages

http://yann.lecun.com/
Note: the best way to reach me is by email or through Hong (I don't check my voicemail very often). My main research interests are Machine Learning, Computer Vision, Mobile Robotics, and Computational Neuroscience. I am also interested in Data Compression, Digital Libraries, the Physics of Computation, and all the applications of machine learning (Vision, Speech, Language, Document understanding, Data Mining, Bioinformatics).
http://www.cs.colostate.edu/~whitley/
From 1997 to 2002 Prof. Whitley served as Editor-in-Chief for the journal Evolutionary Computation published by MI Press. In 2005 ISGEC became a Special Interest Group (Sigevo) of ACM. In 2007 Prof. Whitley was elected Chair of Sigevo. Research interests Genetic Algorithms,Neural Networks, Local Search, Elementary Landscapes, Scheduling Applications, Theoretical Foundations of Genetic Algorithms. Publications and Biographical Information
http://domino.research.ibm.com/comm/research_people.nsf/pages/rshankar.index.html
PhD in Computer Science from the University of São Paulo (USP). Disciplinas 2010-1 Compiladores | Programação Research interests Machine learning (especially unsupervised learning, online learning), one-class classification, novelty detection, concept drift, natural computing and bio-inspired computing (especially evolutionary computation, genetic programming, genetic algorithms and artificial neural networks),

have authored documents related to t and other closely-related topics. Similarly, if an author, a has expertise on a topic $t \in T$, authors similar to a could be expected to write about t and topics related to t.

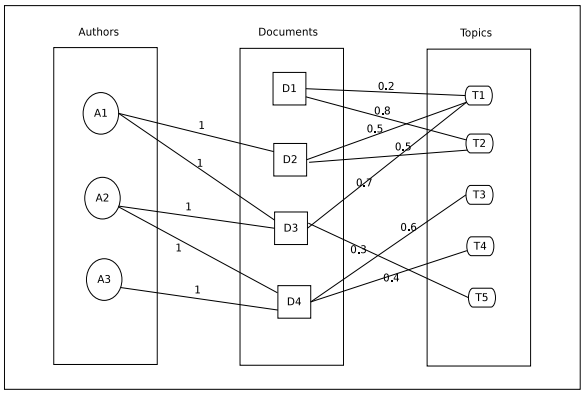


Fig. 5. An example Author-Document-Topic (ADT) graph

The associations between documents and their authors and documents and their topics can be represented by a weighted tri-partite graph as follows: Let

$G = (V, E)$ represent such a graph where the vertex set, $V = A \cup D \cup T$ is the union of author, A, document, D and topic nodes, T. Edges between A and D reflect the authorship relation between documents and authors whereas edges between D and T reflect the topical association of documents. Weights assigned to the edges in ADT capture the association strength between two nodes. For instance, the edges between document and topic nodes can be assigned weights using the proportion of that topic in the document.

Table 6. Top expert recommendations using ADT models in CiteSeer^x.

Natural Language Processing	Machine Learning	Information Retrieval	Semantic Web
Hermann Ney	Raymond J. Mooney	W. Bruce Croft	Ian Horrocks
Aravind K. Joshi	Vasant Honavar	Douglas W. Oard	Dieter Fensel
Raymond J. Mooney	Manuela Veloso	Hermann Ney	Enrico Motta
Bonnie J. Dorr	Jude Shavlik	Jamie Callan	Amit Sheth
Alex Waibel	David B. Leake	Hector Garcia-molina	Steffen Staab

Table 7. Top "similar expert" recommendations using ADT models in CiteSeer^x

Christopher D. Manning	Tom M. Mitchell	W. Bruce Croft	James Hendler
Aravind K. Joshi	Raymond J. Mooney	Douglas W. Oard	Ian Horrocks
Martha Palmer	Sebastian Thrun	Jamie Callan	Dieter Fensel
Raymond J. Mooney	Peter Stone	Justin Zobel	Amit Sheth
Timothy Baldwin	Jude Shavlik	Norbert Fuhr	Frank Van Harmelen
Bonnie J. Dorr	Vasant Honavar	Maarten De Rijke	Wolfgang Nejdl

An example ADT graph is shown in Figure 5. We refer the reader to [20] for details regarding scoring author nodes using this graph and show for illustration some of the anecdotal examples included in this paper. The top-5 author recommendations obtained for sample topic and name queries in Computer Science using the ADT graph generated from the CiteSeer^x collection provided in [20] are shown in Tables 6 and 7. As the presented examples illustrate, topic models provide insights into the document collections that be incorporated for learning specific extraction and ranking tasks in digital libraries.

5 Summary and Conclusions

In this tutorial, we summarized some common tasks in digital libraries and presented automated techniques based on our own research experiences in CiteSeer^x, an open-access, digital library portal. In particular, we described a few unsupervised, weakly-supervised, and semi-supervised techniques for performing metadata extraction and classification tasks related to research documents in Computer Science and related areas. Based on the experimental results provided in the referenced papers, we conclude that techniques combining machine learning algorithms and domain-specific insights yield models that perform competitively

on several tasks which once involved intense human labor. We hope the presented techniques provide an overview of the challenges for applying machine learning research to specific retrieval and extraction tasks in a large, practical system and possible solutions for addressing the same.

References

1. The literature of bibliometrics, scientometrics, and informetrics. *Scientometrics*, 52(2):291–314, 2001.
2. David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.
3. Florian Boudin. A comparison of centrality measures for graph-based keyphrase extraction. In *IJCNLP*, 2013.
4. Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.*, 2(2):121–167, June 1998.
5. Cornelia Caragea, Jian Wu, Kyle Williams, Sujatha Das Gollapalli, Madian Khabsa, Pradeep Teregowda, and C. Lee Giles. Automatic identification of research articles from crawled documents. In *Web-Scale Classification: Classifying Big Data from the Web, co-located with WSDM*, 2014.
6. Soumen Chakrabarti. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan-Kauffman, 2002.
7. Soumen Chakrabarti, Martin van den Berg, and Byron Dom. Focused crawling: a new approach to topic-specific web resource discovery. *Comput. Netw.*, 31(11-16):1623–1640, May 1999.
8. Bi Chen, Leilei Zhu, Daniel Kifer, and Dongwon Lee. What is an opinion about? exploring political standpoints using opinion scoring model. In *AAAI*, 2010.
9. Isaac G. Councill, C. Lee Giles, and Min-Yen Kan. Parscit: an open-source crf reference string parsing package. In *LREC*, 2008.
10. Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990.
11. Hongbo Deng, Irwin King, and Michael R. Lyu. Formal models for expert finding on dblp bibliography data. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 163–172, Washington, DC, USA, 2008. IEEE Computer Society.
12. Gregory Druck, Gideon Mann, and Andrew McCallum. Learning from labeled features using generalized expectation criteria. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 595–602, New York, NY, USA, 2008. ACM.
13. Mohamed Firdhous. Automating legal research through data mining. *CoRR*, abs/1211.1861, 2012.
14. Eibe Frank, Gordon W. Paynter, Ian H. Witten, Carl Gutwin, and Craig G. Nevill-Manning. Domain-specific keyphrase extraction. In *IJCAI*, 1999.
15. Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. Posterior regularization for structured latent variable models. *J. Mach. Learn. Res.*, 11:2001–2049, August 2010.
16. Sujatha Das Gollapalli and Cornelia Caragea. Extracting keyphrases from research papers using citation networks. In *AAAI*, pages 1629–1635, 2014.

17. Sujatha Das Gollapalli, Cornelia Caragea, Prasenjit Mitra, and C. Lee Giles. Researcher homepage classification using unlabeled data. In *Proceedings of the 22nd international conference on World Wide Web*, WWW '13, pages 471–482, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.
18. Sujatha Das Gollapalli, C. Lee Giles, Prasenjit Mitra, and Cornelia Caragea. On identifying academic homepages for digital libraries. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, JCDL '11, pages 123–132, New York, NY, USA, 2011. ACM.
19. Sujatha Das Gollapalli, Prasenjit Mitra, and C. Lee Giles. Learning to rank homepages for researcher-name queries. In *SIGIR Workshop on Entity Oriented Search*, 2011.
20. Sujatha Das Gollapalli, Prasenjit Mitra, and C. Lee Giles. Ranking experts using author-document-topic graphs. In *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '13, pages 87–96, New York, NY, USA, 2013. ACM.
21. Sujatha Das Gollapalli, Yanjun Qi, Prasenjit Mitra, and C. Lee Giles. Extracting researcher metadata with labeled features. In *SDM*, pages 740–748, 2014.
22. Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, April 2004.
23. KhaledM. Hammouda, DiegoN. Matute, and MohamedS. Kamel. Corephrase: Keyphrase extraction for document clustering. In *Machine Learning and Data Mining in Pattern Recognition*. 2005.
24. Hui Han, C. Lee Giles, Eren Manavoglu, Hongyuan Zha, Zhenyue Zhang, and Edward A. Fox. Automatic document metadata extraction using support vector machines. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, JCDL '03, pages 37–48, Washington, DC, USA, 2003. IEEE Computer Society.
25. Jiawei Han. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., 2005.
26. Taher Haveliwala, Sepandar Kamvar, Dan Klein, Chris Manning, and Gene Golub. Computing pagerank using power extrapolation. Number 2003-45. Stanford, 2003.
27. Qi He, Bi Chen, Jian Pei, Baojun Qiu, Prasenjit Mitra, and C. Lee Giles. Detecting topic evolution in scientific literature: how can citations help? In *CIKM*, pages 957–966, 2009.
28. Gregor Heinrich. Parameter estimation for text analysis. Technical report, 2008.
29. Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '99, pages 50–57, New York, NY, USA, 1999. ACM.
30. A. Hulth. Improved automatic keyword extraction given more linguistic knowledge. *EMNLP*, pages 216–223, 2003.
31. Aleks Jakulin, Wray Buntine, Timothy La Pira, and Holly Brasher. Analyzing the u.s. senate in 2003: Similarities, clusters, and blocs. *Political Analysis*, 17(3):10, June 2009.
32. Steve Jones and Mark S. Staveley. Phrasier: A system for interactive document retrieval using keyphrases. In *SIGIR*, 1999.
33. Saurabh Kataria, Krishnan S. Kumar, Rajeev Rastogi, Prithviraj Sen, and Srinivasan H. Sengamedu. Entity disambiguation with hierarchical topic models. In *KDD*, pages 1037–1045, 2011.

34. Saurabh Kataria, Prasenjit Mitra, and Sumit Bhatia. Utilizing context in generative bayesian models for linked corpus. In *AAAI*, 2010.
35. Saurabh Kataria, Prasenjit Mitra, Cornelia Caragea, and C. Lee Giles. Context sensitive topic models for author influence in document networks. In *IJCAI*, pages 2274–2280, 2011.
36. Su Nam Kim and Min-Yen Kan. Re-examining automatic keyphrase extraction approaches in scientific articles. In *Proceedings of the Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications*, MWE '09, 2009.
37. SuNam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. Automatic keyphrase extraction from scientific articles. 47(3), 2013.
38. John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.
39. Huajing Li, Isaac G. Councill, Levent Bolelli, Ding Zhou, Yang Song, Wang-Chien Lee, Anand Sivasubramaniam, and C. Lee Giles. Citeseerx: a scalable autonomous scientific digital library. In *Proceedings of the 1st international conference on Scalable information systems*, InfoScale '06, New York, NY, USA, 2006. ACM.
40. Xiaoli Li, See-Kiong Ng, and Jason T. L. Wang. *Biological Data Mining and Its Applications in Healthcare*. World Scientific Publishing Co., Inc., 1st edition, 2013.
41. Bing Liu. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications)*. Springer-Verlag New York, Inc., 2006.
42. Feifan Liu, Deana Pennell, Fei Liu, and Yang Liu. Unsupervised approaches for automatic keyword extraction using meeting transcripts. In *Proceedings of NAACL '09*, pages 620–628, 2009.
43. Xiaoyong Liu and W. Bruce Croft. Statistical language modeling for information retrieval. *ARIST*, 39(1):1–31, 2005.
44. Gideon S. Mann and Andrew McCallum. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *J. Mach. Learn. Res.*, 11:955–984, March 2010.
45. Gideon S. Mann and Andrew McCallum. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *J. Mach. Learn. Res.*, 11:955–984, March 2010.
46. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.
47. Luís Marujo, Ricardo Ribeiro, David Martins de Matos, João Paulo Neto, Anatole Gershman, and Jaime G. Carbonell. Key phrase extraction of lightly filtered broadcast news. *CoRR*, 2013.
48. ThuyDung Nguyen and Min-Yen Kan. Keyphrase extraction in scientific publications. In *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers*, volume 4822. 2007.
49. Jos-Luis Ortega-Priego, Isidro F. Aguillo, and Jos Antonio Prieto-Valverde. Longitudinal study of contents and elements in the scientific web environment. *Journal of Information Science*, 32(4):344–351, 2006.
50. Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, 1999.
51. Nirmala Pudota, Antonina Dattolo, Andrea Baruzzo, Felice Ferrara, and Carlo Tasso. Automatic keyphrase extraction and ontology mining for content-based tag recommendation. *Int. J. Intell. Syst.*, 2010.

52. Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., 1986.
53. Sunita Sarawagi. Information extraction. *Found. Trends databases*, 1(3):261–377, March 2008.
54. Jie Tang, Ruoming Jin, and Jing Zhang. A topic modeling approach and its integration into the random walk framework for academic search. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 1055–1060, Washington, DC, USA, 2008. IEEE Computer Society.
55. Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 990–998, New York, NY, USA, 2008. ACM.
56. Pradeep B. Teregowda, Isaac G. Councill, R. Juan Pablo Fernández, Madian Khabsa, Shuyi Zheng, and C. Lee Giles. Seersuite: Developing a scalable and reliable application framework for building digital libraries by crawling the web. In *Proceedings of the 2010 USENIX Conference on Web Application Development*, WebApps'10, 2010.
57. Suppawong Tuarob, Line C. Pouchard, and C. Lee Giles. Automatic tag recommendation for metadata annotation using probabilistic topic modeling. In *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '13, pages 239–248. ACM, 2013.
58. Jian Wu, Kyle Williams, Hung-Hsuan Chen, Madian Khabsa, Cornelia Caragea, Alexander Ororbia, Douglas Jordan, and C. Lee Giles. Citeseerx: Ai in a digital library search engine. In *IAAI*, 2014.
59. Hongyuan Zha. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In *SIGIR*, 2002.
60. Shuyi Zheng, Ding Zhou, Jia Li, and C. Lee Giles. Extracting author meta-data from web using visual features. In *Proceedings of the Seventh IEEE International Conference on Data Mining Workshops*, ICDMW '07, pages 33–40, Washington, DC, USA, 2007. IEEE Computer Society.