Learning and visualizing topics and their change with time for the exploratory analysis of social tags and multilingual topic modeling of chemical compounds

Der Naturwissenschaftlichen Fakultät III der
Martin-Luther Universität Halle-Wittenberg
zur Erlangung des akademischen Grades eines
Dr. rer. nat.

eingereichte Dissertation

von
Herr Dipl.-Bioinform. André Gohr
aus
Potsdam

Datum der Einreichung: 2012/05/15

Referent: Doz. Dr. Alexander Hinneburg, Martin-Luther Universität Halle-Wittenberg, 06099 Halle, Deutschland
Koreferent: Prof. Dr. Stefan Wrobel, Rheinische Friedrich-Wilhelms Universität Bonn, 53012 Bonn, Deutschland

Tag der mündlichen Prüfung: 2012/12/19

Contents

Abstract

I propose AdaptivePLSA for dynamic topic modeling with streams of documents. For the SIGIR proceedings, the learned topics give clear hints to the main research subjects. Next, I propose TopicTable, a visualization for presenting topics learned from document streams. TopicTable visualizes useful pieces of information, e.g., topics similarities and newly emerging words. It is effective as it provides clear hints to alien documents which were added to a test stream of documents. Next, I propose an approach for the disambiguation of social tags which have been added to documents by many users of a collaborative tagging system. This approach uncovers unobvious semantics of tags and visualizes topics which are learned from the tagged documents. Last, I apply bilingual topic modeling to NMR spectra and chemical constitutions of chemical compounds. The learned bilingual topics might be exploited by new approaches for data mining in chemical- and structure-databases of chemical compounds.

Keywords: probabilistic topic models, data mining, visualization, collaborative tagging systems, analysis of social tags, statistical modeling of chemical data, dynamic topics, statistical modeling of document streams, statistical modeling of 2D NMR spectra, statistical modeling of chemical constitutions, chemical databases, chemical structure databases, bilingual topic models, PLSA, Probabilistic Latent Semantic Analysis, LDA, statistical modeling of document collections, statistical modeling of social tags, social tags, disambiguation for social tags, unobvious semantics of social tags, semantic analysis of social tags

Zusammenfassung

Ich schlage AdaptivePLSA für das Lernen von dynamischen Topics aus Dokumentströmen vor. Für die SIGIR Konferenzbände liefern die gelernten Topics Hinweise auf die wissenschaftlichen Hauptthemen. Ich schlage TopicTable als eine Visualisierung für die aus Dokumentströmen gelernten Topics vor. TopicTable visualisiert nützliche Zusatzinformationen wie Topicähnlichkeiten und neu auftretende Wörter. In einem Beispiel liefert TopicTable eindeutige Hinweise auf fremdartige Dokumente in einem Dokumentstrom. Desweiteren beschäftige ich mich mit dem Aufdecken der semantischen Mehrdeutigkeit von sozialen Tags. Der vorgestellte Ansatz deckt unerwartete Bedeutungen dieser Tags auf und visualisiert Themen der Dokumente mit diesen Tags. Zuletzt wende ich ein bilinguales Topic-Modell an, um NMR-Spektren und chemische Konstitutionen chemischen Verbindungen zu modellieren. Die gelernten bilingualen Topics könnten Anwendung finden in neuartigen Ansätzen zum Datamining in chemischen Strukturdatenbanken.

Schlagwörter: Probabilistische Themenmodelle, Datamining, Visualisierung, kollaborative Tagging-Systeme, Tag-Analyse, statistische Modellierung chemischer Daten, dynamische Topics, statistische Modellierung von Dokumentströmen, statistische Modellierung von 2D-NMR-Spektren, statistische Modellierung von chemischen Konstitutionen, chemische Datenbanken, Strukturdatenbanken, bilinguale Themenmodelle, PLSA, Probabilistic Latent Semantic Analysis, LDA, statistische Modellierung von Dokumentmengen, statistische Modellierung von sozialen Tags, Begriffsklärung für soziale Tags, soziale Tags, unerwartete Semantik sozialer Tags, semantische Analyse sozialer Tags

Acknowledgments

I deeply thank the many people who have assisted and promoted me in accomplishing this dissertation. First of all, I thank Prof. Wessjohann, Dr. Porzel and, Dr. habil. Hinneburg for ceding the topic of this thesis to me. I especially thank Prof. Wessjohann, an expert of chemoenzymatics and synthesis, for his endorsement, for valuable discussions, and his openness to computational approaches to biochemical questions. I especially thank Dr. Porzel for her assistance, input of ideas, advices on NMR spectroscopy, and for the warm-hearted working atmosphere. Next, I especially thank Dr. habil. Hinneburg, Prof. Spiliopoulou, and Dr. Schult, for many valuable discussions, many helpful advices and valuable input especially on new ideas, for their assistance and for the productive collaboration. Further on, I very much appreciate the helpful discussions with Dr. Tilo Lübken about NMR spectroscopy and modeling NMR data, and the valuable support of Ricardo Usbeck in context of our collaboration on tag sensemaking. Special thanks go to Kathi and Mark, my collaborators at the Leibniz Institute of Plant Biochemistry, for their encouragement, very good collaboration and friendship. Last but not least, I deeply thank Claudia, Bernhard, Henning, Ilja and Rajnish for their support and for all the good moments which I could share with them.

Chapter 1

Introduction

From cuneiform writing to instant messaging, written texts have constituted an important part of human communication from the very beginning. Increasing literacy turned written texts into a tool of the masses. Ever since, large parts of the population have, actively and passively, participated in written communication. For several thousand years by now, mankind has been aware of texts as one form of recording human knowledge. Consequently, we started to systematically collect and categorize our knowledge and to put it into encyclopedic books about 2000 years ago.

A first revolution of publishing is the invention of mechanical printing of books, i.e., woodblock printing and the letterpress invented by the Chinese (ca. 8^{th} century) and Johannes Gutenberg (15^{th} century), respectively. With the dawn of personal computers (PC), which became available for household use in the 1980's, a second revolution of writing texts began. Writing and publishing texts became simpler, and, with the progression of the world wide web (WWW), new ways of publishing like online news, electronic mail, and instant messaging emerged. Nowadays, PCs and the WWW provide a large number of people around the globe with the possibility of writing and publishing texts electronically. The dominance of written texts has changed the quality of human communication. This strongly became evident in the scientific field, where the number of journals steadily increased from the first two modern journals founded in 1665 [1], i.e., the French *Le Journal des Sçavans* and the Philosophical Transactions published in London, to about 23750 in the year 2006 [2]. Similarly, as visualized in Figure 1.1, the number of scientific articles published per year has dramatically increased; from an estimated number of 344 in 1726 to about 1.5 million in 2009 [3]. Nowadays, electronically published texts have become the most important way of scientific communication, e.g., in math, engineering and natural sciences.

On the one hand, the enormous amount of electronic documents are a vast source of knowledge. On the other hand, effective managing and searching for relevant documents in these enormous volumes is challenging. Computers are not only a tool for writing and publishing but also for managing, organizing and retrieving information from large text collections. For example, nowadays we often use online search-engines and links between documents for a keyword-guided search. But, the combination of computers with modern approaches of machine learning like topic modeling could even make more powerful, thematic-guided orientation, overviewing, managing, and searching in large volumes of documents possible. David Blei describes thematic-guided work as follows ([4], p. 1):

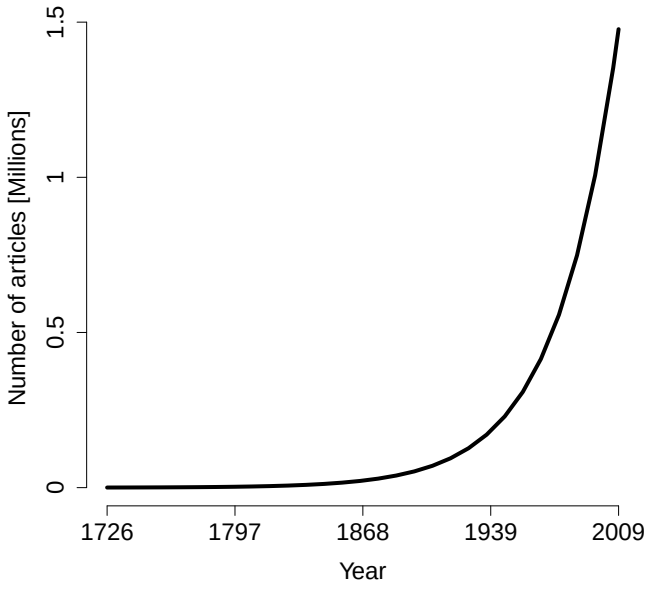


Figure 1.1: Number of published research articles per year estimated by Jinha [3]. Their cumulative sum reaches about 50 million by the end of 2009.

> "Imagine searching and exploring documents based on the themes that run through them. We might 'zoom in' and 'zoom out' to find specific or broader themes; we might look at how those themes changed through time or how they are connected to each other. Rather than finding documents through keyword search alone, we might first find the theme that we are interested in, and then examine the documents related to that theme."

Probabilistic topic models were proposed in the community of machine learning in the 1990's [4]. Most topic models assume the word distributions of documents to be a mixture of relatively few discrete prototype distributions over the whole vocabulary. These prototype distributions are called topics. Intuitively, a topic represents a combination of words which often occur together in documents. These combinations often can be interpreted by humans as thematic subjects of the documents. Meaningfully interpretable topics learned in an unsupervised manner from documents are a potential basis for thematic-guided orientation, overviewing, managing and searching in large volumes of documents.

Exploiting the effectiveness of topic models to learn meaningful topics is a general approach taken in this thesis for solving different problems. In more detail, this thesis presents the following research subjects. In Chapter 3, we shall get familiar with an extension of the learning process for the topic model called Probabilistic Latent Semantic Analysis (PLSA) from batch to online learning. Online-learning of PLSA is learning topics and their change with time from a stream of documents. Learning topics over time is useful in the context of studying document streams. When interpreting the learned topics and their changes, the reader might get an idea about the present thematic subjects and their lifespans. Further on, having discovered a thematic subject of interest, the reader might go into more detail by reading documents related to this subject.

Moreover, studying the content of a stream might also be useful for other objectives. For example, by learning topics from a stream of Twitter[1] messages, Twitter users and staff will get an thematic overview of the kind of subjects users are most interested in during a certain period of time. The extension of PLSA to online learning was published at the SIAM Conference on Data Mining in 2009 [5].

In Chapter 4, we shall get familiar with a visualization technique for presenting topics and their evolution over time. Often, users are primarily interested in studying the learned topics, which basically are distributions over the entire vocabulary consisting of up to several thousand words. Hence, applying topic models for studying topics is only as effective as the presentation of the learned topics is. The proposed visualization technique called TopicTable is especially tailored to presenting topics learned from document streams. It presents changes of word patterns, topic strength, topic evolution and similarities among topics over time. TopicTable might be used to present topics that have been learned by any topic model that is capable of learning topics and their evolution over time. This work was published at the International Conference on Knowledge Discovery and Information Retrieval (KDIR) 2010 where the submitted paper won the Best Student Paper Award [6]. An extension of this paper was selected for publication in the series Communications in Computer and Information Science [7]. In addition, results of the KDIR paper and future prospects were presented in the special session on visual analytics of the LWA[2] forum 2011 of the German Computer Science Society [8].

In Chapter 5, we shall see how online learning of PLSA and the visualization technique might be applied to discover meanings of tags that are used in collaborative tagging systems. Collaborative tagging systems are online systems for managing resources, e.g., pieces of music[3] or bibliographic references[4]. Users upload resources and associate them with tags, which are publicly visible, short, descriptive words or phrases. For instance, the bibliographic reference entitled "Toward a Generalized Bayesian Network" [9] has tags[5] *bayes*, *model-building* and *para-leer*. Meanings of some tags might be less obvious to a specific reader because tagging is an unsupervised, public activity; all users might choose arbitrary words as tags and attach these to all resources at will. The here proposed approach is meant as help for exploring the meanings of tags. Exploring and understanding the variety of possible meanings of tags, users might work more effectively with social tagging systems. A part of this work was published in combination with work on TopicTable [6]. Work on detecting unobvious semantic meanings of social tags was published at the International Conference on Web Intelligence, Mining and Semantics 2011 [10].

In Chapter 6, we shall see how effective a polylingual extension of PLSA is for modeling chemical data. Topic models have been primarily proposed for studying documents by learning topics from these. Their effectiveness of learning topics might be helpful for other research directions, too. In this work, the polylingual extension of PLSA is applied to the prediction of fragments of the chemical constitution of chemical compounds from

[1] `twitter.com`, May 3, 2012

[2] LWA stands for *Lernen, Wissen, Adaption*, which in English means Learning, Knowledge, Adaptation. The LWA forum took place in Magdeburg, Germany, September 28–30, 2011, `http://lwa2011.cs.uni-magdeburg.de` (May 3, 2012)

[3] `last.fm`, May 3, 2012

[4] `citeulike.org`, May 3, 2012

[5] `citeulike.org`, January 29, 2012

their nuclear magnetic resonance (NMR) spectra and vice versa. In a real-world application, such predicted structural fragments might be used as fingerprints of chemical compounds. These fingerprints might then be used for look-ups in chemical databases, e.g., to search for structurally similar compounds. Parts of this work was presented at the International Conference on Machine Learning 2010 [11].

Before more details of the mentioned works are presented, general concepts of topic modeling shall be discussed in Chapter 2. In this chapter, we shall learn what topic models are, what a topic exactly is, and what differentiates topic models from mixture models. Further on, this chapter presents details of Bayesian learning for PLSA and explains the relation of PLSA to the most popular topic model Latent Dirichlet Allocation. Last, a toy example of learning topics from documents is presented which shows how the learned topics agree with the prior knowledge about the thematic subjects of the toy documents.

Chapter 2

Probabilistic topic modeling

Basic concepts and notation often used in context of probabilistic topic modeling shall be discussed in this chapter. First, a common representation of documents is introduced. Next, the mixture of unigrams, which is a simple topic model, and Probabilistic Latent Semantic Analysis (PLSA), the basic topic model used in this thesis, are described. Afterwards, a parameter prior for PLSA is defined before the Expectation Maximization algorithm, which is used for parameter learning, is discussed. The close relation between PLSA and the popular topic model Latent Dirichlet Allocation is made clear and, last, PLSA is exemplarily applied to some toy documents and the learning topics are analyzed.

2.1 Document representation as co-occurrence data

Topics are learned from a given set of documents, which is called a corpus of documents. In this corpus each document has its own unique document ID d. The number of different document IDs is denoted by N and the document IDs range from 1 to N. Some documents might be identical to each other although they have different document IDs. Documents are made up of occurrences of words from a given vocabulary. The vocabulary spans M different words, each of which is identified by a unique word ID w with $1 \leq w \leq M$.

The word occurrences in documents are described by pairs (d, w) of a document ID and a word ID. Such a pair encodes that the word with ID w occurs once in the document with ID d. The vector of pairs, one for each word occurrence in each document, $\vec{X} = ((d_1, w_1), \ldots, (d_{|\vec{X}|}, w_{|\vec{X}|}))$ describes the given corpus. The number of pairs, i.e., the number of elements of vector $\vec{X}$, is denoted by $|\vec{X}|$ and, for compactness of writing, a pair (d_i, w_i) is also denoted by $(d, w)_i$. If a word occurs several times in a document, then the corresponding pair occurs several times in $\vec{X}$. So, the length $|\vec{X}|$ is equal to the total number of word occurrences in the given corpus.

As no correspondence between the order of pairs in vector $\vec{X}$ and the order of words in documents is enforced, information about the word order is lost. This is inline with many topic models that often do not model the order of word occurrences in documents. Instead, they assume each document to be a *bag of words*, i.e., a *set* of word occurrences. Examples of topic models that assume documents to be bags of words are the mixture of unigrams, Probabilistic Latent Semantic Analysis, and Latent Dirichlet Allocation.

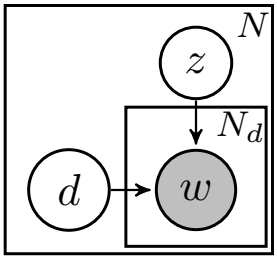


Figure 2.1: Plate-model representation of the mixtures of unigrams. This plate model visualizes the generative process of a corpus of N documents; the d^{th} document consists of N_d word occurrences.

2.2 Mixtures of unigrams

The term *n-gram* refers to a sequence of n subsequent words in a document. A n-gram of length $n = 1$ is referred to as a unigram. The mixture of unigrams is a simple probabilistic topic model.

Mixture models approximate complex distributions as combinations of several simpler distributions, which are called component distributions. A mixture of unigrams is a mixture of $K \geq 2$ generalized Bernoulli distributions, which are indexed by the index variable $1 \leq z \leq K$. Each generalized Bernoulli distribution $P(w|z)$ with $1 \leq w \leq M$ is a discrete distribution over the vocabulary.

The mixture of unigrams approximates the word distribution of each document by exactly one component distribution. In other words, it assumes that all words of one document are drawn from one component distribution [12]. Under this model, a document is generated by drawing a document ID d first. Then, a component distribution is chosen by drawing a component index z. Last, the words of the document d are sampled from the component distribution $P(w|z)$. Figure 2.1 shows a plate model of this generative process.

Usually, it is unknown from which components the words of the documents were sampled; the component indices are hidden variables. Hence, under the mixture model the probability of each pair (d, w) is given by marginalization over the hidden variable.

$$P(w, d) = P(d)P(w|d) \tag{2.1}$$

$$= P(d) \sum_{k=1}^{K} P(w|z = k)P(z = k) \tag{2.2}$$

The underlying statistical independence assumptions are (i) once the component index z is known, the distribution of words does not depend on the document d anymore, i.e., $P(w|z, d) = P(w|z)$, and (ii) the component probability is independent of the document d, i.e., $P(z|d) = P(z)$. As a consequence, the weights or prior probabilities of each component distribution $P(z)$ with $1 \leq z \leq K$ are specific for the entire corpus.

Component distributions encode combinations of words that often co-occur in different documents. The words of such a combination will get a relatively high probability in the component distribution that corresponds to this combination.

2.3 Topics – patterns of co-occurring words

Patterns of co-occurring words are combinations of words that often co-occur in documents. These patterns reflect word-to-word relatedness. Two fundamental ways of determining word-to-word relatedness, different from exact keyword matching, are [13]

1. compute semantic correlations between words of the vocabulary
2. compute frequency–co-occurrence statistics of words from large corpora

When taking the approach of probabilistic topic modeling, we learn a number K of patterns of co-occurring words from co-occurrence statistics in an unsupervised manner. These patterns are called topics and mathematically they are conditional, generalized Bernoulli distributions over the discrete space defined by the vocabulary. Generalized Bernoulli distributions are also known under the name multinomial distribution. Topics are indexed by an index variable $1 \leq z \leq K$ and the word probabilities of each topic sum up to 1.

$$\forall 1 \leq z \leq K \quad : \quad 1 = \sum_{m=1}^{M} P(w = m|z) \tag{2.3}$$

Each topic reflects a pattern of co-occurring words. Words of the encoded pattern have a relatively high conditional probability given that topic. Having learned topics from a given document corpus and studying the most likely words per topic, humans are often able to interpret these as thematic subjects of the investigated documents. For example, the words plane, airport, crash, flight, safety, aircraft, passenger, board, airline might indicate a subject about aircraft (example from [14]).

Learning a mixture of unigrams involves learning its K component distributions $P(w|z)$ with $1 \leq z \leq K$. These distributions are topics and they are learned under the assumption that words of each document are sampled from a single topic. But this assumption is unlikely to be fulfilled by real documents as these are often mixtures of thematic subjects themselves. Thus, it comes as no surprise that mixtures of unigrams have been shown to be ineffective for probabilistically modeling of large document corpora [15].

The assumption of documents being mixtures of thematic subjects is the fundamental idea behind admixture models, of which Probabilistic Latent Semantic Analysis and Latent Dirichlet Allocation are prominent examples. Before Probabilistic Latent Semantic Analysis is introduced in more detail, the following section gives clues about why admixture models are effective for topic modeling.

2.4 Admixture models

Although mixtures of unigrams are effective in modeling complex word distributions, they still have limitations. If each document is a mixture of topics itself, the mixture of unigrams could fail to learn these word patterns. An illustrative example is modeling the heights of people in three cities: Tokyo, Stockholm, and Berlin. We assume that three different classes of tallness exist: small, medium and tall, and that the same number of people live in each city. The heights of people who are small is equally distributed over a certain range of tallness that is visualized by a red bar in Figure 2.2. The same is

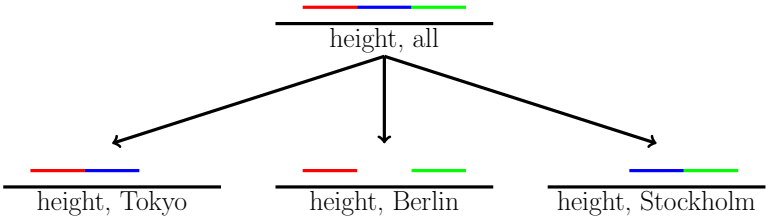


Figure 2.2: Distribution of heights of people in three cities. Diagram at the top illustrates the distribution of heights when observing all people at once.

true for the medium and tall heights; the corresponding ranges are visualized by a blue, and a green bar, respectively. People in Tokyo are small and medium in tallness, people in Berlin are small and tall, and people in Stockholm are medium and tall. Taking the mixture model approach for learning a distribution over the heights of people, we would investigate the heights of all people from all cities at once without exploiting the information about from which cities the people come. But, studying the heights of all people at once makes the discovery of the three classes of tallness nontrivial: these heights are equally distributed across the whole range of tallness from small to tall (top of Figure 2.2).

Admixture models are more sophisticated and were firstly proposed in genetics [16]. In context of genetics, an admixture arises when two previously isolated populations begin interbreeding [17]. As a consequence, *each* offspring is genetically a *mixture* of the genome of the isolated populations. An admixture topic model assumes *each* document to be a *mixture* of patterns of co-occurring words, i.e., topics. This assumption fits our intuition about documents better; patterns of co-occurring words might be different in different parts of a document, e.g., the introduction or conclusions of scientific papers. In other words, the valuable assumption of an admixture topic model is that documents themselves are mixtures of topics.

Returning to the illustrative example and focusing at the bottom of Figure 2.2, one might get an idea why an admixture model might be better suited for learning the three classes of tallness. The admixture model exploits the information about which people come from which cities. The admixture assumption here is that the distribution of heights in each city is a mixture of some common classes of height. We might clearly observe the two classes small and tall when we observe people from Berlin. The different classes of height in Tokyo and Stockholm are still not visible directly. A crucial capability of admixture models is helpful here: information about the latent classes is mediated among the different cities. For example, the clear distinction between small and tall people from Berlin is helpful for discovering that the distribution of height of people from Stockholm indeed is a mixture of the two classes tall and medium. This is why, in the case of topic models, modeling a document corpus with an admixture model might result in more meaningful topics as compared to those which might be obtained with a simple mixture model approach.

2.5 Probabilistic Latent Semantic Analysis

Probabilistic Latent Semantic Analysis (PLSA) is an admixture topic model; it assumes each document to be a mixture of topics. By that, PLSA overcomes the limitation of the mixture of unigrams, which assumes each document to be represented by only one topic.

2.5.1 Generative process

PLSA was proposed by Hofmann in 1999 [14, 18]. PLSA consists of K topics, which are generalized Bernoulli distributions and which are indexed by the index variable $1 \leq z \leq K$. Words of each document are sampled from a document-specific mixture of these topics. The weights of this mixture for document d are given by the parameters of a generalized Bernoulli distribution $P(z|d)$, which is a discrete distribution over indices 1 to K specific for document d.

The generation of a corpus of documents as described by pairs of document and word IDs $\vec{X}$ is assumed to be as follows. Each pair $(d, w)_i$ with $1 \leq i \leq |\vec{X}|$ is drawn by the following three steps.

$$d_i \sim P(d) \tag{2.4}$$
$$z_i \sim P(z|d = d_i) \tag{2.5}$$
$$w_i \sim P(w|z = z_i) \tag{2.6}$$

First, the document ID d_i is drawn from a generalized Bernoulli distribution, which is a discrete distribution over document IDs from 1 to N. Then, a topic index z_i is sampled from the document-specific distribution over topic indices. Last, a word ID is drawn from the topic with index z_i.

The fundamental assumption underlying this generative process is that for a given pair (d, w) the word ID and document ID are conditionally independent of each other if the topic is known. In more detail, the joint probability $P(d_i, w_i, z_i)$ can always be factorized as $P(w_i|z_i, d_i)P(z_i|d_i)P(d_i)$. If the topic index z_i is known, then word w_i will be sampled from that topic independently of the document d_i. Consequently, the conditional probability $P(w_i|z_i, d_i)$ depends only on z_i but not on d_i and one obtains $P(d_i, w_i, z_i) = P(w_i|z_i)P(z_i|d_i)P(d_i)$.

2.5.2 Parametrization and likelihood

The parameters of a PLSA model are the following:

1. parameters $\vec{\delta}$ of the generalized Bernoulli distribution $P(d)$ that encodes document probabilities

2. for each document: parameters $\vec{\theta}_n$ of the generalized Bernoulli distribution $P(z|d = n)$ that encodes the document-specific topic weights called *topic-mixture proportions*

3. for each topic: parameters $\vec{\omega}_k$ of the generalized Bernoulli distribution $P(w|z = k)$ that encodes the topic-specific word probabilities called *word-topic associations*

All parameters of a PLSA model ζ are jointly denoted by $\zeta := (\vec{\theta}, \vec{\omega}, \vec{\delta})$.

In the following, the PLSA parameters are specified in more detail. The variables n, k, and w lie in the following ranges: $1 \leq n \leq N$, $1 \leq k \leq K$, and $1 \leq m \leq M$. The parameter $\vec{\theta}$ is defined as $\vec{\theta} := (\vec{\theta}_1, \ldots, \vec{\theta}_N)$ with $\vec{\theta}_n := (\theta_{n,1}, \ldots, \theta_{n,K})$ and $\theta_{n,k} := \log P(z = k|d = n)$[1]. Consequently, the *topic-mixture proportions* of the n^{th} document are given by $(\exp(\theta_{n,1}), \ldots, \exp(\theta_{n,K}))$. The parameter $\vec{\omega}$ is defined as $\vec{\omega} := (\vec{\omega}_1, \ldots, \vec{\omega}_K)$ with $\vec{\omega}_k := (\omega_{k,1}, \ldots, \omega_{k,M})$ and $\omega_{k,m} := \log P(w = m|z = k)$. The *word-topic associations* of the k^{th} topic are given by $(\exp(\omega_{k,1}), \ldots, (\exp \omega_{k,M}))$. The parameter $\vec{\delta}$ is defined as $\vec{\delta} := (\delta_1, \ldots, \delta_N)$ with $\delta_n := \log P(d = n)$. So, the document probability of the n^{th} document is given by $\exp(\delta_n)$.

Pairs of document IDs and word IDs are identically and independently distributed given the model parameters ζ. The topic index variables are usually unobserved for a given document corpus. Consequently, topic indices are marginalized by summing over their states. The likelihood reads as follows.

$$P(\vec{X}|\zeta) = \prod_{i=1}^{|\vec{X}|} P(d_i, w_i|\zeta) \tag{2.8}$$

$$= \prod_{i=1}^{|\vec{X}|} P(d_i|\zeta) \sum_{k=1}^{K} P(w_i|z_i = k, \zeta) P(z_i = k|d_i, \zeta) \tag{2.9}$$

$$= \prod_{i=1}^{|\vec{X}|} \exp(\delta_{d_i}) \sum_{k=1}^{K} \exp(\omega_{k,w_i}) \exp(\theta_{d_i,k}) \tag{2.10}$$

2.5.3 Prior

Hofmann proposed PLSA without defining a prior over model parameters [14, 18]. In contrast to Hofmann, who took the Maximum-Likelihood approach for learning PLSA, a Bayesian approach is taken in this thesis. Bayesian learning requires a prior over model parameters. Generally, priors might enhance learning of parameters of statistical models in two ways.

1. They express a-priori knowledge about the parameters in a mathematical, principled manner.

2. Priors may smooth parameter estimates. If data sets are small in size, smoothing is especially useful to prevent artifacts of parameter learning like zero probabilities.

[1] PLSA invokes probability mass functions of the generalized Bernoulli distribution. In general, a generalized Bernoulli distribution is defined for a random variable Y that might take discrete values as encoded by the integers $1, \ldots, S$. Its natural parameter is the vector of logarithmic probabilities $\vec{\eta} = (\eta_1, \ldots, \eta_S) = (\log P(Y = 1), \ldots, \log P(Y = S))$, with $\sum_{s=1}^{S} \exp(\eta_s) = 1$. The generalized Bernoulli distribution assigns the following probability to each realization of Y

$$\forall 1 \leq s \leq S: \quad P(y = s|\vec{\eta}) := \exp\left(\sum_{s=1}^{S} \eta_s \mathbf{1}_{y=s}\right) \tag{2.7}$$

with $\mathbf{1}$ denoting the Kronecker delta.

The prior for parameter $\vec{\delta}$ is a uniform prior. In the context of Bayesian learning, defining a uniform prior is equivalent to not defining a prior at all, and so the prior for parameter $\vec{\delta}$ is omitted in the following. Priors for $\vec{\theta}$ and $\vec{\omega}$ are independent of each other. Eventually, the prior over the PLSA parameters reads as

$$P(\zeta) \;=\; P(\vec{\theta})P(\vec{\omega})\,. \tag{2.11}$$

Prior for word-topic associations

The prior for the parameter $\vec{\omega}$ is a product of K Dirichlets, one Dirichlet for each topic $\vec{\omega}_k$.

$$P(\vec{\omega}) \;=\; \prod_{k=1}^{K} \mathrm{Dir}(\vec{\omega}_k|\beta) \tag{2.12}$$

$$\mathrm{Dir}(\vec{\omega}_k|\beta) \;\propto\; \prod_{m=1}^{M} \exp(\omega_{k,m})^{\frac{\beta}{KM}} \tag{2.13}$$

These Dirichlets have been transformed[2] in correspondence with the transformation of the parameters into the logarithmic space. Normalization factors are neglected for the sake of simplicity. The Dirichlets have one hyper-parameter β which determines the exponents of elements of $\vec{\omega}$ by the principle of equivalent sample-size [19]. This means, the Dirichlet hyper-parameter reflects a-priori data points whose number should be the same when, e.g., comparing PLSA models learned with different vocabularies and numbers of topics. Hence, β is fixed and divided by the number of elements of $\vec{\omega}$, i.e., $K \cdot M$.

Prior for topic-mixture proportions

The prior for topic-mixture proportions is a product of N Dirichlets, one Dirichlet for each document-specific topic-mixture proportions.

$$P(\vec{\theta}) \;=\; \prod_{n=1}^{N} \mathrm{Dir}(\vec{\theta}_n|\alpha) \tag{2.14}$$

$$\mathrm{Dir}(\vec{\theta}_n|\alpha) \;\propto\; \prod_{k=1}^{K} \exp(\theta_{n,k})^{\frac{\alpha}{K}} \tag{2.15}$$

Again, the Dirichlet has one hyper-parameter α; for determining the exponents for elements of $\vec{\theta}_n$, the hyper-parameter α is divided by the number of topics K. It is not divided by $N \cdot K$ as this prior should be independent of the overall number of documents.

[2] As a result, the exponent of $\exp(\omega_{k,m})$ misses the -1 term known from the standard Dirichlet.

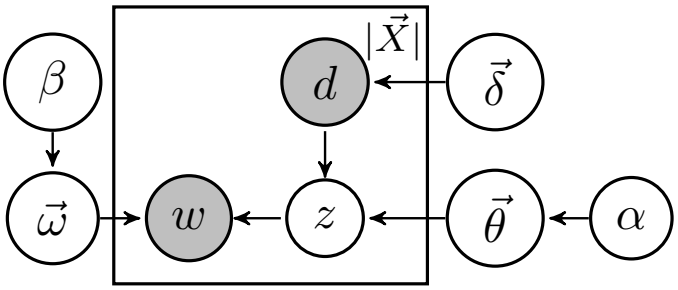


Figure 2.3: Plate-model representation of the generative process underlying PLSA.

2.5.4 Extended generative process

The generative process (Equations 2.4 - 2.6) is extended by drawing parameters $\vec{\omega}$ and $\vec{\theta}$ from their priors before the data are sampled.

$$\forall 1 \leq n \leq N: \quad \vec{\theta}_n \sim \mathrm{Dir}(\vec{\theta}_n|\alpha) \tag{2.16}$$

$$\forall 1 \leq k \leq K: \quad \vec{\omega}_k \sim \mathrm{Dir}(\vec{\omega}_k|\beta) \tag{2.17}$$

Figure 2.3 shows a graphical representation of the entire generative process using the plate-model representation.

2.5.5 Geometric interpretation in terms of dimension reduction

The word simplex of dimension M is the space defined by points $\vec{x} = (x_1, \ldots, x_M)$ that fulfill (i) $\forall 1 \leq m \leq M : x_m \geq 0$, and (ii) $\sum_{m=1}^{M} x_m = 1$. In this word simplex documents are represented by points that correspond to their empirical word distributions. An example with three documents is shown in Figure 2.4(a).

The simple unigram model assumes that words of all documents are drawn from a common corpus-specific generalized Bernoulli distribution $P(w)$. This corpus-specific word distribution can be represented by a point of the word simplex, too. A geometrical interpretation of Maximum-Likelihood learning of this corpus-specific word distribution is as follows: we determine the mean of all points that correspond to the empirical word distributions of the documents of a given corpus. A unigram model approximates the word distributions of the documents by this mean distribution. By that, it reduces the space of word distributions of documents to this single point.

The mixture of unigrams assumes that all words of each document are drawn from one out of K corpus-specific topics. These topics again are word distributions and can be represented as K points $\exp(\vec{\omega}_k)$, $1 \leq k \leq K$, of the word simplex. Figure 2.4(b) illustrates an example with $K = 3$ topics. The mixture of unigrams approximates the word distribution of each document to exactly one topic. In other words, the points that correspond to the word distributions of the documents are mapped to one of the topic points. Hence, the space of word distributions is reduced to the K topic points.

Similar to the mixture of unigrams, PLSA assumes K topics. These K points define the basis of a sub-space of the word simplex which is called topic simplex. This topic simplex is geometrically defined by all points that are linear combinations of the K topic points where the mixture weights sum up to 1. In other words, each point of the topic simplex is a word distribution that is a mixture of the K word-topic associations $\exp(\vec{\omega}_k)$.

Being an admixture model, PLSA approximates the word distribution of each document by a linear combination of the K topics. Geometrically, this means that the points of the word simplex that correspond to the word distributions of the given documents are mapped somewhere onto the topic simplex. Hence, the space of word distributions of the documents is reduced to the topic simplex.

To make an example, PLSA consists of topic-mixture proportions for all documents. The learned topic-mixture proportions $(\exp(\theta_{n,1}), \ldots, \exp(\theta_{n,K}))$ of the n^{th} document define a point in the K-dimensional topic simplex. The word distribution of this document is approximated by a document-specific mixture of the K topics: $\sum_{k=1}^{K} \exp(\vec{\omega}_k) \exp(\theta_{n,k})$. Figure 2.4(c) illustrates how word distributions are approximated by mapping their points from the word simplex onto the topic simplex. Figure 2.4(d) shows an example topic simplex; its points correspond to the topic-mixture proportions of the learned PLSA model.

The unigram model, the mixture of unigrams, and PLSA all map the empiric word distributions of documents onto a sub-space of the word simplex. In comparison to the other two models, PLSA might better capture the approximated word distributions as PLSA has a larger statistical expressiveness. Instead of mapping the empirical word distributions of the documents to a few single points of the word simplex, it maps these somewhere onto a sub-space of the word simplex, namely onto the topic simplex

2.6 Parameter learning for PLSA

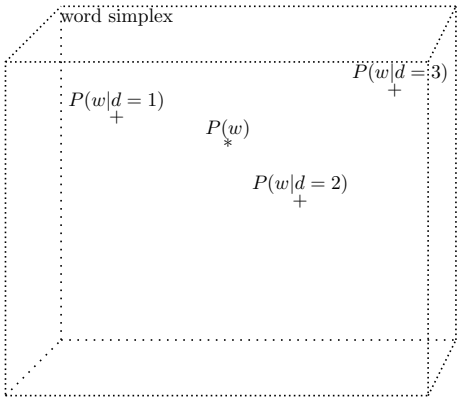
Hofmann [14, 18] followed the Maximum-Likelihood principle for learning the parameters of a PLSA model. Inline with Cien and Wu [20], I proposed to apply the principle of Maximum-A-Posteriori (MAP) to parameter learning of PLSA [5]. In section 2.6.1, the derivation of the Expectation Maximization algorithm for MAP parameter learning is presented. Afterwards, different approaches for determining an optimal number K of topics are discussed in Section 2.6.2. The relation between PLSA and the popular Latent Dirichlet Allocation is subject of the Section 2.6.3.

2.6.1 EM algorithm for MAP learning

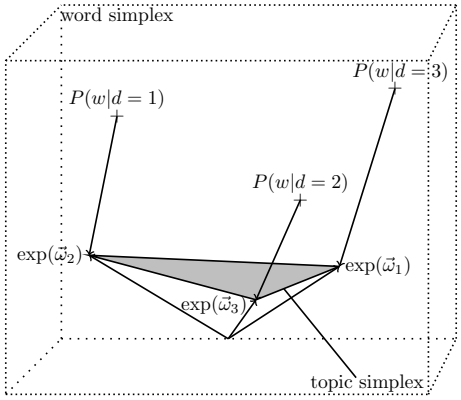
The data are pairs of document and word IDs $(d, w)_i$ with $1 \leq i \leq |\vec{X}|$. PLSA assumes that the word w_i in document d_i is drawn from one of the K topics. The index of this topic is given by the index variable $1 \leq z_i \leq K$. The topic-index variables for all pairs are denoted by $\vec{Z} = (z_1, \ldots, z_{|\vec{X}|})$.

As the values of the topic indices $\vec{Z}$ are unknown, the marginalized a-posteriori probability of the model parameters must be maximized.

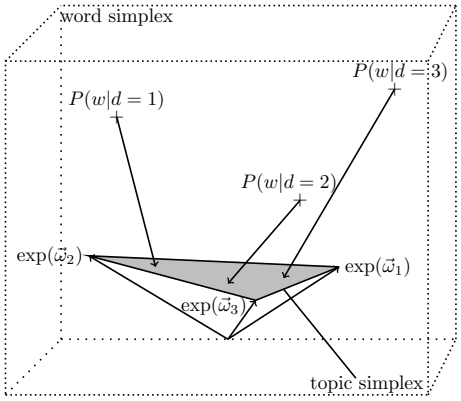
$$\begin{aligned} \vec{\zeta}^* &:= \operatorname*{argmax}_{\zeta} P(\zeta|\vec{X}) && (2.18) \\ &= \operatorname*{argmax}_{\zeta} \sum_{\vec{z}} P(\zeta, \vec{Z} = \vec{z}|\vec{X}) && (2.19) \\ &= \operatorname*{argmax}_{\zeta} \sum_{\vec{z}} P(\vec{X}, \vec{Z} = \vec{z}|\zeta) P(\zeta) && (2.20) \end{aligned}$$



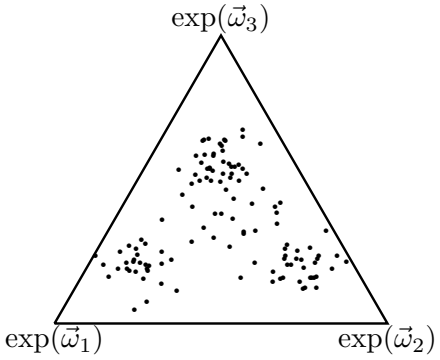
(a) Unigram



(b) Mixture of unigrams



(c) PLSA, topic simplex embedded in word simplex



(d) PLSA, top-view on topic simplex

Figure 2.4: Geometric interpretation of unigram model, mixture of unigrams, and PLSA.

This maximization is solved with the help of the Expectation Maximization (EM) algorithm [21], which is a sophisticated gradient-ascent approach for parameter learning of probabilistic models with hidden variables [22–24]. The EM algorithm iteratively re-estimates parameters in order to locally maximize the a-posteriori probability $P(\zeta|\vec{X})$. In each iteration, the algorithm goes through two steps: the expectation (E) and the maximization (M) step. The parameter estimates in iteration t are denoted by $\zeta^{(t)}$.

E step

During the E step of the $(t+1)^{\text{th}}$ iteration, the algorithm computes posterior probabilities for each hidden variable z_i, $1 \leq i \leq |\vec{X}|$, and for each of its states $1 \leq k \leq K$.

$$\gamma_{i,k}^{(t+1)} := P(z_i = k|\zeta^{(t)}, \vec{X}) \tag{2.21}$$

Applying the Bayes rule, one finds that these posteriors are proportional to the product of the likelihood and $P(z_i = k|\zeta^{(t)})$.

$$\gamma_{i,k}^{(t+1)} \propto P((d,w)_i|z_i = k, \zeta^{(t)})P(z_i = k|\zeta^{(t)}) \tag{2.22}$$

The posteriors are computed using the parameter estimates $\zeta^{(t)}$ of the last M step of EM iteration t. The probability $P(z_i = k|\zeta^{(t)})$ is identified with the topic probability $P(z = k)^{(t)}$ for topic k in the last EM iteration.

$$\gamma_{i,k}^{(t+1)} = \frac{P((d,w)_i|z_i = k, \zeta^{(t)})P(z = k)^{(t)}}{\sum_{\bar{k}=1}^{K} P((d,w)_i|z_i = \bar{k}, \zeta^{(t)})P(z = \bar{k})^{(t)}} \tag{2.23}$$

$$= \frac{\exp(\omega_{k,w_i}^{(t)}) \exp(\theta_{d_i,k}^{(t)}) P(z = k)^{(t)}}{\sum_{\bar{k}=1}^{K} \exp(\omega_{\bar{k},w_i}^{(t)}) \exp(\theta_{d_i,\bar{k}}^{(t)}) P(z = \bar{k})^{(t)}} \tag{2.24}$$

All posteriors determined in the $(t+1)^{\text{th}}$ iteration are jointly denoted by $\vec{\gamma}^{(t+1)}$.

M step

The model parameters are re-estimated by analytically maximizing the expectation of the sum of the log-complete-data-likelihood and the log-prior. The expectation is determined with respect to the posteriors $\vec{\gamma}^{(t+1)}$.

$$(\zeta^*)^{(t+1)} := \underset{\zeta^{(t+1)}}{\operatorname{argmax}} \, \mathbb{E}\left[\log P(\vec{X}, \vec{Z}|\zeta^{(t+1)}) + \log P(\zeta^{(t+1)})\right]_{P(\vec{Z}|\zeta^{(t)},\vec{X})} \tag{2.25}$$

The following parameter estimates ($1 \leq k \leq K$, $1 \leq m \leq M$, $1 \leq n \leq N$) result from this maximization.

$$\exp(\omega_{k,m}^*)^{(t+1)} \propto {}^{\beta}/_{KM} + \sum_{\substack{1 \leq i \leq |\vec{X}|, \\ w_i = m}} \gamma_{i,k}^{(t+1)} \tag{2.26}$$

$$\exp(\theta_{n,k}^*)^{(t+1)} \propto {}^{\alpha}/_{K} + \sum_{\substack{1 \leq i \leq |\vec{X}|, \\ d_i = n}} \gamma_{i,k}^{(t+1)} \tag{2.27}$$

For compactness of writing, Normalization constants, which would result from the constraints $1 =: \sum_{m=1}^{M} \exp(\omega_{k,m}^*)^{(t+1)}$, and $1 =: \sum_{k=1}^{K} \exp(\theta_{n,k}^*)^{(t+1)}$, were neglected in the Equations above.

Document probabilities $\vec{\delta}$ are estimated without the EM algorithm as these are independent of the hidden variables $\vec{Z}$. Because of the uniform prior, MAP estimates of the document probabilities for documents $1 \leq n \leq N$ are equal to the relative document lengths

$$\exp(\delta_n^*) \propto N_n \tag{2.28}$$

with N_n being the number of word occurrences in the n^{th} document.

The probability of topic k in the EM iteration $(t+1)$ is computed as follows.

$$P(z = k)^{(t+1)} := \frac{\sum_{i=1}^{|\vec{X}|} \gamma_{i,k}^{(t+1)}}{\sum_{\bar{k}=1}^{K} \sum_{i=1}^{|\vec{X}|} \gamma_{i,\bar{k}}^{(t+1)}} \tag{2.29}$$

This topic probability is not a model parameter but needed by the EM algorithm.

Break condition

The EM algorithm continuously goes through E and M steps until some break condition stops it. Common quantities for break conditions are the number of EM iterations already passed or the improvement of the a-posteriori probability between two successive EM iterations.

Local maxima

Getting stuck in local optima or saddle points is a risk of gradient-ascent methods like the EM algorithm. An option of dealing with this problem is to re-start the EM algorithm several times and to use different start configurations of the model parameters each time. Afterwards, one would omit all but the very EM run that reached the highest a-posteriori probability. The last parameter estimates of this run are then used to instantiate a PLSA model.

2.6.2 Number of topics

The number of topics K is an essential parameter of topic models. Often, the optimal number of topics depends on the specific application at hand. If general topics are of interest, the number of topics should be substantially smaller than the number of documents, i.e., $K \ll N$. The number of topics has an impact on their quality and, hence, on their interpretation. For some applications, users might fix the number of topics prior to learning the topic model. For other applications it might be necessary to determine the optimal number of topics by some kind of evaluation strategy for assessing different numbers of topics.

Fixing the number of topics prior to learning

Generally, the number of topics influences the detailedness of learned patterns of word co-occurrences. Figure 2.5 visualizes the general tendency of the expected detailedness of topics according to their number. If the number of topics is small, then learned topics will reflect coarse patterns of co-occurring words probably via higher-order co-occurrences. Higher-order co-occurring words are those whose occurrences are linked via a third or a fourth word; for example, word w_2 connects occurrences of separately occurring words w_1 and w_3 because w_2 co-occurs with w_1 and with w_3 [25]. If the number of topics increases, then learned topics will encode finer patterns of co-occurring words, from more specific subjects to single phrases or even to single words (actually, in this case one would not call this pattern a co-occurrence pattern).

The expected detailedness of learned topics could be a reasonable guidance for fixing the number of topics prior to learning a topic model. This strategy could be effective, e.g., for an exploratory analysis by which one wants to uncover general thematic subjects that are present in a large corpus.

Determining the number of topics during learning

If fixing the number of topics prior to learning is inexpedient, one might determine an optimal number of topics by some evaluation strategy. The number of topics could be

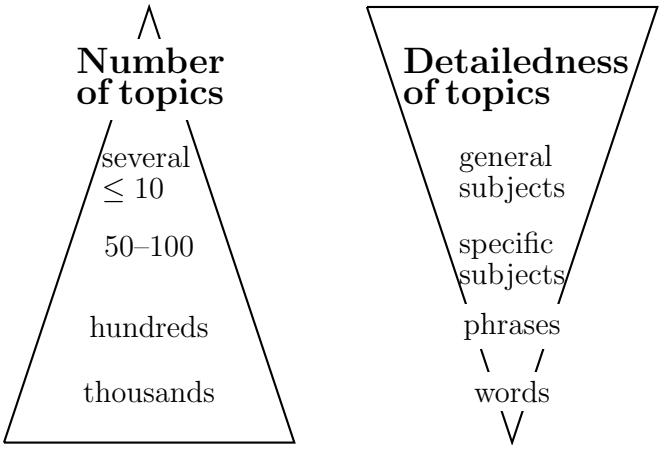


Figure 2.5: Relationship between number of topics and detailedness of captured patterns of co-occurring words.

assessed by the goodness-of-fit of the entire topic model that has been learned with different numbers of topics. Often, the goodness-of-fit of a topic model is measured by the likelihood or the perplexity of yet unseen data [25]. Other measures for assessing the goodness-of-fit of a topic model might be deduced from a specific application at hand. For example, if a topic model is used for classifying emails into *ham* and *spam*, then the observed classification performance reached for different numbers of topics could be used to determine an optimal number of topics. Another approach is to measure the interpretability of the learned topics [26]. This approach assesses the topics themselves instead of using some kind of computational score.

Different evaluation procedures could be used for the purpose of determining the goodness-of-fit. Examples include cross-validation with help of which a topic model is learned with a subset of the data and the goodness-of-fit is determined with respect to the remaining data. As a result, one obtains empiric estimates of the mean goodness-of-fit and its variance.

A very different approach for dealing with the uncertainty about the optimal number of topics is to extend topic models such that they take into account a distribution over the number of topics instead of considering a fixed number of topics. An example of this approach is the extension of a topic model by the Dirichlet process, which models a distribution over the number of topics [27]. Although this approach makes determining an optimal number of topics obsolete, it comes with the need to determine an optimal distribution over the number of topics. Different parameter setting of the Dirichlet process, for example, might significantly affect the goodness-of-fit of the resulting topic model.

2.6.3 Bayesian learning

Although parameter learning following the MAP principle takes parameter priors into account, it determines point estimates of parameters. A fully Bayesian approach treats parameters as hidden variables with prior distributions and aims at determining their posterior distribution. Often, this posterior is not tractable directly. In this case, methods that approximate the posterior, e.g., Variational Bayes (VB) [28, 29], are applicable.

Another approach would be to use Markov Chain Monte Carlo methods [30, 31], e.g., Gibbs Sampling (GS), which simulate samples from the posterior.

In 2003, Blei et al. proposed Latent Dirichlet Allocation (LDA) [15]. The generative processes of LDA and the extended generative process of PLSA (as given in Section 2.5) are almost the same [32]. A slight difference is that PLSA directly uses document IDs but LDA not. Asuncion et al. [32] showed that the generative process of LDA can be extended by incorporating document IDs. The close relatedness between PLSA and LDA have been discussed by other researchers, too. Kaban et al. [33] stated that the combination of PLSA and Maximum-Likelihood parameter learning, as originally proposed by Hofmann [14, 18], essentially reveals the Maximum-Likelihood solution of LDA. In a later work, Buntine et al. [34] also discussed the relatedness between PLSA and LDA. As the generative processes of PLSA and LDA are closely related, the essential difference between both models is the approach taken for parameter learning.

Different approaches for parameter learning have been suggested for LDA. Initially, Blei et al. proposed VB; later, Griffiths et at. proposed GS [35]. Refinements of these procedures have been proposed in the sequel; for example, collapsed VB [36] or collapsed GS [37, 38].

Often, fully Bayesian approaches are assumed to be superior over MAP point estimates of model parameters. The seminal work of Asuncion et al. [32] sheds light on how MAP parameter learning for PLSA is related to VB or GS for LDA. Comparing PLSA and LDA is possible because both models assume almost the same generating process. Beside other methods for parameter inference, Asuncion et al. compared MAP parameter learning for PLSA, and VB, collapsed VB, GS and collapsed GS for LDA. Asuncion et al. found that the observed differences between PLSA and LDA with respect to, e.g., the perplexity of unseen data, might be eliminated by optimal choices of the hyper-parameters. Asuncion et al. [32] state: "... our empirical results suggest that these inference algorithms have relatively similar predictive performance when the hyper-parameters for each method are selected in an optimal fashion." Hence, if hyper-parameters are chosen optimally, then a PLSA model whose parameters have been learned with MAP is comparable, with regard to the predictive performance, to LDA whose parameters have been inferred with VB or GS.

Beside prediction, topic models are often used for exploratory, unsupervised data analyses to uncover thematic subjects present in the given documents. For such unsupervised analyses estimates of the model parameters, i.e., topic-mixture proportions and word-topic associations, are helpful. Asunction et al. [32] showed that estimates of topic-mixture proportions and word-topic associations following the MAP principle are very similar to those which have been obtained by VB. In summary, these findings suggest that (i) the generative process of PLSA is very similar to that one of LDA, and (ii) the MAP principle for parameter learning and the VB approach are equally useful for parameter learning.

2.7 Example study for PLSA

An exploratory analysis of a small document corpus is presented in this section. This toy example gives clues about the capabilities of PLSA, e.g., how topics learned from the documents look like and how well the learned topics agree with the thematic subjects

used for the construction of the toy documents.

The documents are composition of three different subjects: theology, computer, and plant. Each subject is associated with seven words as shown in Table 2.1. The union of all these 21 words defines the vocabulary of the corpus.

Two different corpora of three bunches of 100 documents, which are made up of 20 word occurrences, are considered. Each document of the first corpus was constructed from words of one subject only whereas the documents of the second corpus mainly consisted of words from one subject but contain some words from the other subjects, too.

2.7.1 Thematically pure documents

Each bunch corresponds to one subject. All 100 documents from a bunch were constructed by sampling with equal probability 20 words from all words of the corresponding subject. For example, a document on theology could look like this: spirit worship god god prayer god sin prayer worship creation god trinity creation spirit creation spirit god trinity creation trinity.

Parameters of a PLSA model with $K = 3$ topics were learned following the Maximum-Likelihood principle. The learned topics and averaged topic-mixture proportions are presented in Figure 2.6. Topics are presented by visualizing their discrete cumulative distribution functions over the vocabulary. To this end, words are sorted for each topic according to their probability (word-topic association) in decreasing order.

Inspecting the first learned topic in Figure 2.6(a), we find that all words from theology have a relative high probability whereas all other words are negligibly likely. Thus, we can clearly interpret this topic as being about theology. Studying the word-topic associations for the other two topics, we clearly find that the second/third topic is about computer/plant.

Studying the learned topic-mixture proportions in Figure 2.6(d), we find the first topic, which we have interpreted as being about theology, is most likely in documents of the bunch that was constructed from the theology subject. The second topic (computer) and the third topic (plant) are most likely in documents from the bunch that corresponds by construction to the subject computer and plant, respectively.

Inspecting topic-mixture proportions for each bunch separately, we find that PLSA clearly assigns most probability mass to the topic that corresponds to the true subject of each bunch. Inspecting topic-mixture proportions averaged over the whole corpus, we find that each topic is equally likely. These findings are consistent with the construction of the corpus; all bunches are of the same size, hence, each topic should be equally likely when all documents are considered at once.

2.7.2 Thematically mixed documents

A fundamental assumption of PLSA is that documents are mixtures of topics. The first corpus is a special case, in which each document is about one thematic subject only. Now, each bunch of 100 documents mainly corresponds to one topic, but documents are mixtures of the three subjects. For each document, ten words were sampled with equal probability from the subject of the bunch to which this document belongs. Another 10 words were sampled with equal probability from the whole vocabulary. A document

Table 2.1: Words assigned to the three subjects.

theology	**computer**	**plant**
god	keyboard	plant
spirit	ram	root
trinity	monitor	leaf
creation	printer	flower
worship	laptop	basal
prayer	usb	seed
sin	internet	soil

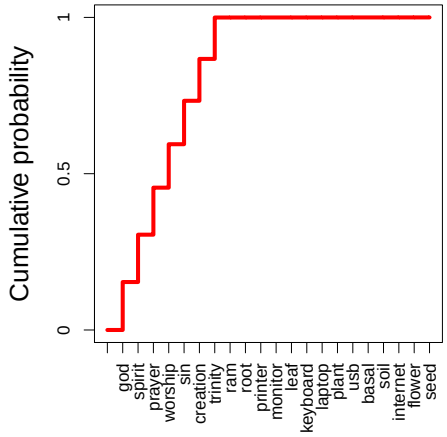
mainly about theology could be: usb flower internet worship monitor root spirit creation trinity god basal monitor spirit creation trinity trinity keyboard keyboard worship trinity.

Again, a PLSA model with $K = 3$ topics was learned. The resulting PLSA parameters are visualized in Figure 2.7. We find all seven words associated to theology, six of all seven words associated to computer, and six words associated to the subject about plant among the top-ten words of topic 1, topic 2, and topic 3, respectively (Figures 2.7(a-c)). Hence, we again can clearly interpret the first, second, and third learned topic as being about theology, computer and plant, respectively.

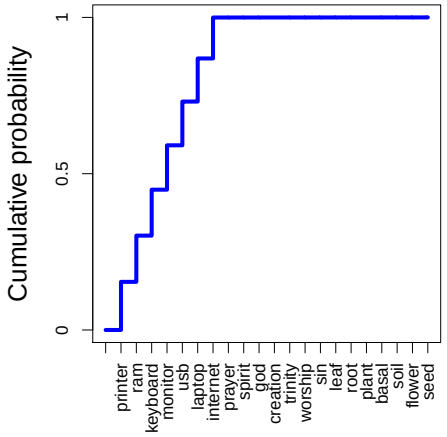
The learned topic-mixture proportions are shown in Figure 2.7(d). Again, we find one learned topic to be dominant in each bunch. Further on, the topic that is dominant in each bunch corresponds to the main subject of this bunch. For example, the first topic, which could be interpreted as being about theology, is the most likely topic in the bunch that corresponds, due to construction, to the subject on theology. As in the previous example, we find that the relative probabilities of the learned topics in each bunch agree with the thematic construction of these bunches. This is also true for the entire corpus; all three topics are almost equally likely as one would expect from the construction of the corpus.

2.7.3 Summary

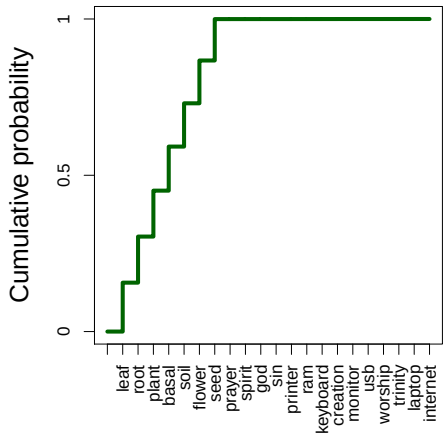
An intuitive example of an exploratory study with PLSA for a document corpus was presented in this section. The three PLSA topics, which have been learned in an unsupervised manner, could be well related to the thematic subjects which were used for the construction of the documents. In addition, the learned topic-mixture proportions in each bunch of thematically similar documents and across the whole corpus agree well with the expected strength of the thematic subjects.



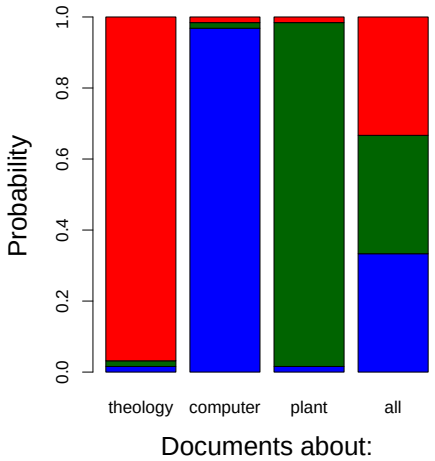
(a) Topic 1 corresponds to thematic subject about theology.



(b) Topic 2 corresponds to thematic subject about computer.

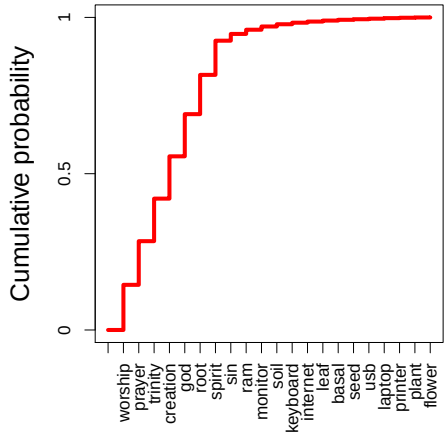


(c) Topic 3 corresponds to thematic subject about plant.

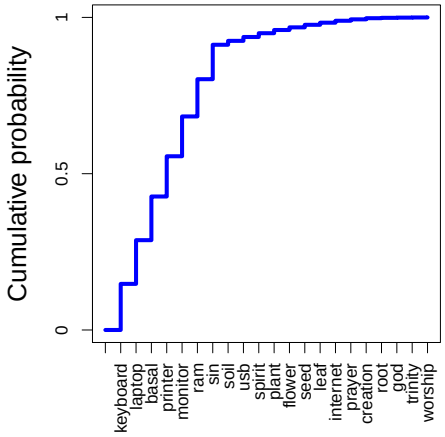


(d) Topic-mixture proportions averaged per bunch and over the whole corpus.

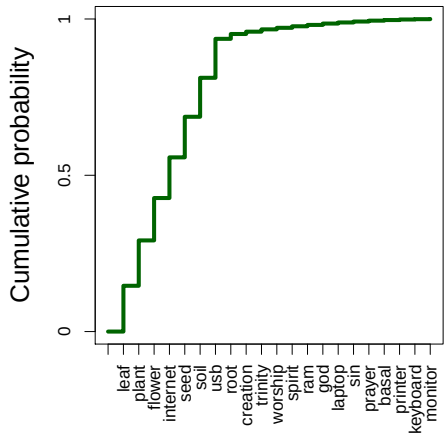
Figure 2.6: Results for thematically pure documents. Colors indicate topics.



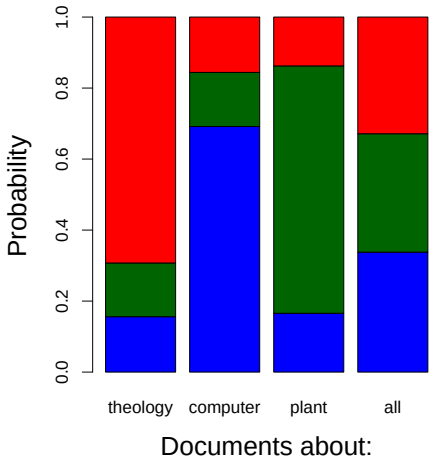
(a) Topic 1 corresponds to thematic subject about theology.



(b) Topic 2 corresponds to thematic subject about computer.



(c) Topic 3 corresponds to thematic subject about plant.



(d) Topic-mixture proportions averaged per bunch and over the whole corpus.

Figure 2.7: Results for thematically mixed documents. Colors indicate topics.

Chapter 3

Learning topics from document streams

Scholars, journalists and practitioners of many disciplines use streaming Web documents for regular acquisition of information on thematic subjects of interest. Examples of streaming documents, which are ordered according to their publication date, are news articles published at the Web, received Twitter messages, and articles published in conference proceedings. Although powerful tools have emerged to assist in this task, they often do not deal with the volatile nature of the Web. Contents of streaming documents may change over time as well as the terminology used in these documents. A changing vocabulary yields static text mining models obsolete and static profile-based filters ineffective for acquisition of information from streaming documents. Text mining models, which take into account publication dates beside contents of documents, have recently attracted a lot attention in the research community. In this chapter, a method for summarizing the changing contents of streaming documents is proposed. This method learns topics and their evolution over time; it discovers topics, adapts them to newly arriving documents, and detects *threads of topics*, while considering the *evolution of terminology*. As such, this method helps in overviewing the contents of streaming documents and in identifying particular thematic subjects of interest.

Most approaches for learning topic evolution can be categorized into methods that treat streaming documents as (i) a finite sequence, or as (ii) a stream of documents. A finite sequence of documents refers to a static collection of documents that are ordered according to their publication date. Methods for finite sequences learn topics from this sequence under the closed-world assumption. Essentially, the whole collection of documents and their time stamps are assumed to be known. The vocabulary over all documents induces a fixed feature space. In contrast, a stream of documents is a collection of documents that continually grows over time. It grows by adding documents in the order of their publication/arrival date. Approaches for streams of documents make the open-world assumption, i.e., yet unknown, new documents arrive, new words emerge and old ones become out-dated. This implies a feature space that changes over time, and, hence, prevents propagation of word statistics backward in time. Word statistics might be propagated forward in time, i.e., the combination of word statistics of old and newly arriving documents are used to update topics.

Stream-based approaches are intuitively closer to the volatile nature of the Web. However, recent evolutionary methods for topic detection based on Probabilistic La-

tent Semantic Analysis [39] and Latent Dirichlet Allocation, e.g., the Dynamic Topic Model [40], and Topics Over Time [41], treat streaming documents as a finite sequence. A simple approach, making the open-world assumption, may take a *retrospective* view of the world: for each new document to be added all information seen so far is used to re-build an extended topic model. This perspective has two disadvantages. First, it only works if the document stream is slow such that retrospection of the complete past is feasible. This disagrees with the conventional perception, according to which each data record is seen only once and processing is limited to whatever information can be accommodated to a comparatively small buffer. This assumption limits the applicability of the approach to those streams, in which the elapsed time between the arrival of any two documents is larger than the time needed to rebuild the model. The assumption holds for slow streams, e.g., all publications of an annual conference but not for fast ones, e.g., an online news-ticker. Second, the feature space grows with time, thus giving raise to the problem of the *curse of dimensionality*: the number of data points needed for a reliable estimate (of any property) grows exponentially with the number of features. Related to the curse of dimensionality is the fact of *data proliferation*, meaning that the feature space grows and, hence, lets grow the demand in space and processing costs with time: very old documents and out-dated words, which are assumed to have, if at all, only a minor impact on current topics, become unnecessary ballast. As learning of most topic models is based on iterative learning algorithms like the EM algorithm, high space demand and processing overhead renders repeated retrospective learning quickly impractical for real application scenarios.

Since a stream cannot be slow and fast at the same time and it will contain future documents with yet unknown words, evolutionary topic monitoring requires adaptation to changes in the word-topic associations and in the vocabulary/feature space. The here proposed method deals with these problems by adapting the feature space *and* the underlying model *gradually*. It applies Probabilistic Latent Semantic Analysis (PLSA) to documents covered by a window that slides across the stream. As the window slides forward, out-dated words and documents are deleted and new documents are incorporated into the model so that the model is adapted to new words and changing contents. This gradual adaption of the model renders retrospection of all past information unnecessary.

Gradual model adaptation brings an inherent advantage over re-learning: not only is the model adapted as a whole, rather each topic learned at some point in time evolves naturally into its follow-up topic. Hence, in contrast to [39], adaptation of PLSA models, each of which learns K topics, results in K *index-based topic threads*, each of which is defined by topics with the same index across all PLSA models. These threads provide a comprehensive summary of the evolution of topics and terminology. Mei and Zhai [39] learn PLSA models individually for each point in time, and so, in order to define threads, they have to match topics of successive PLSA models by some kind of post-processing.

The remainder of this chapter is structured as follows. Related work is discussed in the next section. Then, in Section 3.2, the notations and the sliding window for document streams are introduced. The proposed method for learning topics from document streams is explained in detail in Section 3.3. A baseline approach, to which the proposed method will be compared, is introduced in Section 3.4 before the evaluation framework is explained in Section 3.5. Results of the evaluation are presented in Section 3.6, and the conclusions are given in Section 3.7.

3.1 Related work

Many studies on topic evolution derive topics by clustering of documents; each cluster of documents is assumed to represent one thematic subject. Most of them consider a fixed feature space over the document stream. For example, Morinaga and Yamanishi [42] use an Expectation Maximization approach to build soft clusters. A topic consists of the words with the largest information gain for a cluster. Aggarwal and Yu [43] trace droplets over document clusters. A droplet consists of two vectors that accommodate words associated with the cluster. In contrast, the cluster evolution monitor MONIC [44] and its variants [45–47] treat topic evolution as a special case of cluster evolution over a changing feature space.

Topic models like PLSA [18] and Latent Dirichlet Allocation (LDA) [15] characterize a topic as a generalized Bernoulli distribution over words rather than a cluster of documents. Most of evolutionary topic models based on PLSA or LDA [39–41] make the closed-world assumption. For example, Mei and Zhai [39] assume the complete vocabulary to be known. They learn several PLSA models over time. These are connected by a static unigram model that models the distribution of background words and whose parameters are estimated using all documents of the finite sequence at once. Having learned the PLSA models, Mei and Zhai extract their topics. To define threads of topics, they match topics from different models by applying the KL divergence to the word distributions of the topics.

To find scientific topics from PNAS[1] abstracts, Griffiths and Steyvers [38] learn a single LDA model with all abstracts over time at once. For parameter learning, they use a collapsed Gibbs sampler. They assign temporal properties to topics, such as becoming "cold" or "hot". Incremental LDA [48], which is another approach based on LDA, updates the parameters of the LDA model as new documents arrive, but again assumes a fixed vocabulary and does neither forget the influence of past documents nor of out-dated words.

The dynamic topic model (DTM) [40] learns a sequence of LDA models over time assuming a fixed vocabulary. LDA models are connected by propagating the LDA hyper-parameters forward in time via a state-space model. Similar to Mei and Zhai, topics are extracted from the learned LDA models. A "dynamic topic" is a sequence of topics (word distributions) with the same index over time. AlSumait et al. [49] follow a similar idea and propose Online LDA which propagates hyper-parameters between successive LDA models, too. They extend the Gibbs sampling approach suggested by Griffiths and Steyvers to handle streams of documents. Beside sampling model parameters, they use Gibbs sampling to derive hyper-parameters of topic-word associations of future models. By that, successive LDA models are coupled. AlSumait et al. [49] do not assume a fixed vocabulary; they integrate new words into the model when these first occur. However, AlSumait et al. do not neglect out-dated words. Hence, the vocabulary continually grows and might give rise to the curse of dimensionality.

Wang et al. [41] extend the generative process of LDA in such a way that it additionally generates time stamps of documents. They take a retrospective view: they use the whole finite sequence of documents in combination with their time stamps at once, and assume a fixed vocabulary. A topic over time is a stable distribution over the vocabulary that

[1] Proceedings of the National Academy of Sciences

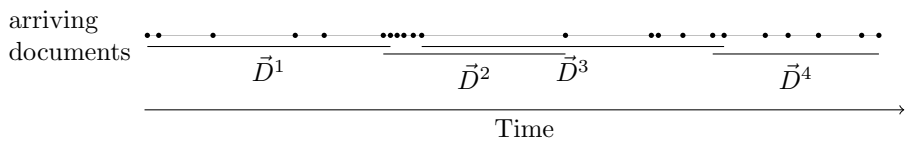


Figure 3.1: A stream of documents. Documents (black dots) arrive at irregular points in time. Horizontal lines represent positions of a sliding window of length $l = 7$ documents. This sliding window shifts forward by $l_{\text{new}} = 5$ documents and it covers at its four positions the following sub-sequences of document IDs $\vec{D}^1$, $\vec{D}^2$, $\vec{D}^3$ and $\vec{D}^4$.

appears at particular points in time. So, Wang et al. learn static topics and, for each topic, periods of time during which this topic is present.

Chien and Wu [20] propose an incremental learning procedure for PLSA. For learning an updated PLSA model for some current documents, they take statistics about all past documents into account. To this end, they exploit the posterior of the last model as the prior for the current model. By this approach Chien and Wu keep all past statistics and so the feature space grows continually with time. In contrast thereto, the here proposed approach neglects word statistics of previous documents immediately when these become out-dated. As such the method for learning topics from document streams, which is proposed in this chapter, neither associates points in time with static topics nor assumes it a static vocabulary. Instead, it learns a sequence of PLSA models by adapting the models to new documents and new words, while removing out-dated documents and obsolete words. Closest to this approach is the Incremental Probabilistic Latent Semantic Indexing method of Chou and Chen [50], which was developed simultaneously to and independently of the here proposed method. In contrast to Chou and Chen, the here described method additionally addresses overfitting by exploiting parameter priors.

3.2 Document streams

A document stream is a data stream with records being documents. A time stamp is assigned to each document. This time stamp refers to the point in time when the document became part of the stream; this event is also called arrival of the document. The time stamp of the n^{th} document is denoted by t_n.

The time stamps induce an ordering of the documents according to increasing arrival date with the special case that no ordering is induced among documents with time stamps that refer to the same point in time. This ordering is represented by the vector of document IDs $\vec{D} = (d_1, d_2, d_3, \ldots)$ with $t_i \leq t_j$ for any $1 \leq i \leq j$. An example stream of documents is depicted in Figure 3.1.

The sequence of successive documents as represented by $\vec{D}$ is split into successive batches with the help of a sliding window. At its i^{th} position the sliding window covers a sub-sequence $\vec{D}^i$ of ordered document IDs from $\vec{D}$. The documents referred to by $\vec{D}^i$ comprise a partial collection of documents ordered according to their time stamp [51]. As a result, the sequence of streaming documents $\vec{D}$ is split into a sequence of successive, possibly overlapping batches: $\vec{D}^1$, $\vec{D}^2$, $\vec{D}^3$, ...

Typically, a sliding window shifts by one document at a time, i.e., the least recent document within the sliding window is forgotten when a new document arrives. In this

thesis, positions of the sliding window are defined either by time or by the number of covered documents as explained in the following.

3.2.1 Definition by time

A sliding window that is defined by time contains all documents that arrive during a certain time interval. If the length of the time interval is u time units and if the window is shifted by u_{new} time units then the window contains at its i^{th} position documents $\vec{D}^i = (d_i, \ldots, d_j)$. The corresponding time stamps $(t_i, \ldots, t_j)$ must lie in the time interval $((i-1)u_{\text{new}}; (i-1)u_{\text{new}} + u]$.

Defining the sliding window via time has two consequences. First, if documents arrive irregularly, then different batches might contain a different number of documents. Second, certain batches may contain no documents at all. A simple way to deal with empty batches is to neglect these during further analyses.

3.2.2 Definition by number of documents

A sliding window which is defined by a number of documents contains l successive documents and shifts by l_{new} documents. That means it shifts to a new position after l_{new} documents have arrived. Such a sliding window covers at position i the following batch $\vec{D}^i = (d_{r(i)}, \ldots, d_{r(i)+l-1})$ with $r(i) = 1 + (i-1)l_{\text{new}}$.

Figure 3.1 visualizes an example with 22 streaming documents. The sliding window covers $l = 7$ and shifts by $l_{\text{new}} = 5$ documents. Four sequential positions of this sliding window are shown by which it defines the batches $\vec{D}^1$, $\vec{D}^2$, $\vec{D}^3$ and $\vec{D}^4$.

Defining the size of the sliding window in document units guarantees that each batch contains the same number of documents. Thus, probabilistic models, which are learned for each batch, are learned from the same number of documents. Two consequences follow from defining the window size in document units. First, at different positions the sliding window may span different periods of time. Second, a larger number of document batches each of which covers a smaller period of time is defined for a fast stream as compared to a slow stream. In this case, learning topics on a more fine-grained time scale with the help of document batches that cover small periods of time might be adequate as topics in a fast stream may change fast.

3.2.3 Dates of sliding window

At its i^{th} position the sliding window defines the batch $\vec{D}^i = (d_i, \ldots, d_j)$. For annotation of the positions of the sliding window/batches, the time stamp of the last covered document is assigned to the window positions/batches. Windows that do not cover any document are left without annotation.

3.3 Adaptive Probabilistic Latent Semantic Analysis

An adaptive learning approach for PLSA, which is called AdaptivePLSA, is presented in this section. This method learns a sequence of PLSA models over time from a stream of documents. For learning a PLSA model, AdaptivePLSA gradually adapts a previous

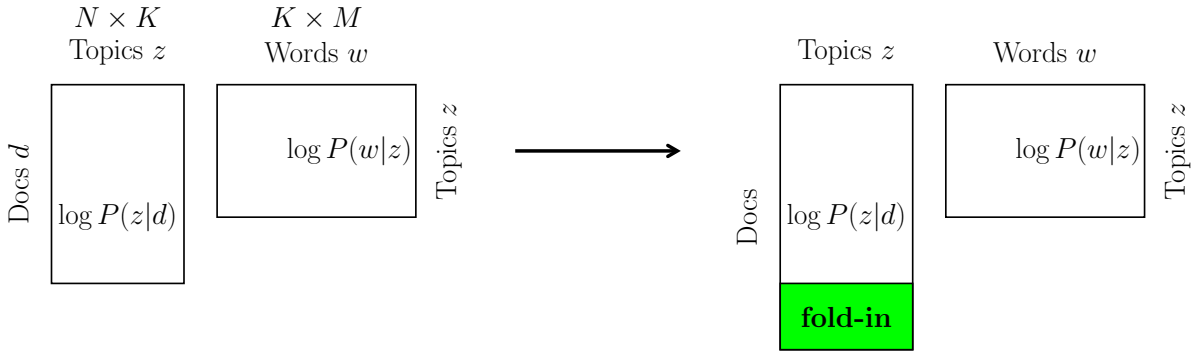


Figure 3.2: Folding-in new documents (determining their topic-mixture proportions) into an already learned PLSA model.

PLSA model to new documents and words. The resulting sequence of adaptively learned PLSA models approximates the contents of the stream of documents. Studying the learned topics, the reader might get an overview of the thematic subjects, which are present in the stream. AdaptivePLSA utilizes new folding-in techniques for gradually learning PLSA models. Parameter learning is done following the principle of Maximum-A-Posteriori as described in Section 2.6.

In the next section, the basic concepts, which are used by AdaptivePLSA, are described. These are *folding-in* of new documents into a PLSA model (Section 3.3.1) and two equivalent parametrizations (word-based and document-based) of a PLSA model (Section 3.3.2). Then, an overview of AdaptivePLSA is presented in Section 3.3.3, and, afterwards, AdaptivePLSA is explained in full depth in Section 3.3.4.

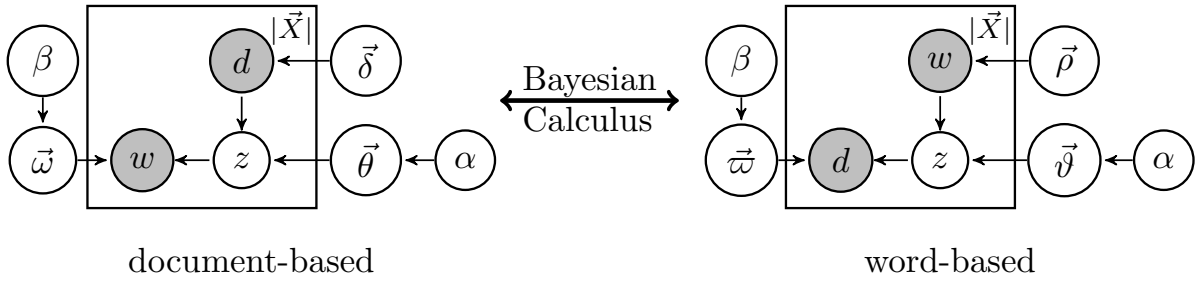
3.3.1 Folding-in documents

As described in Section 2.5, a PLSA model ζ is parametrized by a tuple $(\vec{\theta}, \vec{\omega}, \vec{\delta})$ of logarithmic document probabilities $\vec{\delta}$, logarithmic topic-mixture proportions $\vec{\theta}_d$ for each document d, and logarithmic word-topic associations $\vec{\omega}_k$ for each topic k. The number of topics is denoted by K.

As help for the explanations, a PLSA model is visualized by two matrices as shown in Figure 3.2. Logarithmic topic-mixture proportions are represented by a $N \times K$ matrix; the rows of this matrix contain the logarithmic topic-mixture proportions of the N documents. Logarithmic word-topic associations are represented by a $K \times M$ matrix. The k^{th} row contains the logarithmic word-topic associations of the k^{th} topic.

To incorporate new documents into an already learned PLSA model, Hofmann [18] suggests Maximum-Likelihood folding-in of the new documents. In this thesis, Maximum-Likelihood folding-in is adapted to MAP-folding-in of a new document d'.

The idea behind folding-in is to continue the EM algorithm (Section 2.6.1) having fixed the parameters $\vec{\omega}$ but updating logarithmic topic-mixture proportions $\vec{\theta}_{d'}$ of the new document. In contrast to Maximum-Likelihood folding-in, MAP-folding-in takes the prior of the parameter $\vec{\theta}_{d'}$ into account. For folding-in the new document d' only those word occurrences of it are used that refer to words known by the PLSA model; all other word occurrences are neglected. Hence, folding-in a new document requires that the vocabulary known by the PLSA model and the vocabulary of the new document



document-based word-based

Figure 3.3: Plate-model representation of two equivalent parametrizations of a PLSA model suited for folding-in new documents (left) and for folding-in new words (right). Transformation between both forms is done by Bayesian calculus. N_d denotes the number of word occurrences in document d. N_w denotes the number of document occurrences of word w.

overlap at least to some degree. After having folded-in new documents, AdaptivePLSA is capable of incorporating new words of these documents into the current PLSA model.

Newly determined logarithmic topic-mixture proportions $\vec{\theta}_{d'}$ are concatenated row-wise to the corresponding matrix $[\vec{\theta}\ \vec{\theta}_{d'}]$; this is visualized by the green block in Figure 3.2. Last, document probabilities are re-estimated for all documents (old and new) following Equation 2.28. The parameters of the PLSA model $([\vec{\theta}\ \vec{\theta}_{d'}], \vec{\omega}, \vec{\delta})$ now consist of updated logarithmic topic-mixture proportions and logarithmic document probabilities, and the unchanged logarithmic word-topic associations $\vec{\omega}$.

3.3.2 Document-based and word-based parametrization of PLSA

Two different equivalent parametrizations of a PLSA model are used by AdaptivePLSA. These parametrizations are a consequence of the following two different decompositions of the joint probability $P(d, w)$.

$$P(d,w) = P(d)\sum_{k=1}^{K} P(w|z=k)P(z=k|d) \qquad \text{(document-based)} \tag{3.1}$$

$$= P(w)\sum_{k=1}^{K} P(d|z=k)P(z=k|w) \qquad \text{(word-based)} \tag{3.2}$$

To distinguish these two versions the model corresponding to Equation 3.1 and 3.2 are called *document-based* and *word-based*, respectively. So far, document-based PLSA models have been introduced and discussed. Documents, which are composed of words drawn from K topics that capture word patterns, are the primary focus of document-based PLSA models. In contrast thereto, the main view point of word-based PLSA models is words that are characterized by mixtures of K patterns of document memberships. These patterns are modeled by generalized Bernoulli distributions over document IDs $P(d|z)$ with $1 \leq d \leq M$ and $1 \leq z \leq K$. The patterns of document memberships can be interpreted as combination of documents in which some words often occur together.

In its document-based form a PLSA model is suited for folding-in new documents as topic-mixture proportions can be learned independently of each other. AdaptivePLSA

uses the word-based form for incorporating new words into an already learned PLSA model. This is possible as parameters that corresponds to new words can be learned independently of each other in a similar manner as topic-mixture proportions of new documents in a document-based PLSA model can be learned independently of each other.

Parametrization

Plate models of PLSA models in their document- and word-based form are shown in Figure 3.3. A PLSA model $\zeta = (\vec{\theta}, \vec{\omega}, \vec{\delta})$ in its document-based form consists of logarithmic topic-mixture proportions for documents, logarithmic word-topic associations, and logarithmic document probabilities. A PLSA model $\zeta_w = (\vec{\vartheta}, \vec{\varpi}, \vec{\rho})$ in its word-based form consists of logarithmic topic-mixture proportions for words $\vartheta_{z,w} = \log P(z|w)$, logarithmic document-topic associations $\varpi_{d,z} = \log P(d|z)$, and logarithmic word probabilities $\rho_w = \log P(w)$, with $1 \leq z \leq K$, $1 \leq d \leq N$, $1 \leq w \leq M$. The motivation for using model parameters in the logarithmic domain is that these correspond to the natural parametrization of the underlying generalized Bernoulli distributions (cf. Section 2.5).

Prior

The priors for a document-based PLSA model are described in Section 2.5.3. The same Dirichlet prior as for logarithmic topic-mixture proportions of documents $\vec{\theta}$ (document-based PLSA; cf. Equations 2.14 and 2.15) is used for logarithmic topic-mixture proportions of words $\vec{\vartheta}$ (word-based PLSA). For logarithmic document-topic associations $\vec{\varpi}$ (word-based PLSA) a similar Dirichlet prior is used as for word-topic associations $\vec{\omega}$ (document-based PLSA, cf. Equations 2.12 and 2.13). The difference is that this prior now is defined over the N-dimensional simplex with N denoting the number of documents. Consequently, the hyper-parameter β is divided by $K \cdot N$ instead of $K \cdot M$.

Transformation between document- and word-based PLSA

AdaptivePLSA uses Bayesian calculus for transforming a PLSA model between its document- and word-based form. While explaining AdaptivePLSA in depth, details of transforming PLSA models shall be discussed in Section 3.3.4.

3.3.3 Overview

When a sliding window is applied to a given stream of documents $\vec{D}$, one obtains a sequence of batches $\vec{D}^1, \vec{D}^2, \ldots$. AdaptivePLSA is applied to these batches from which it learns a sequence of PLSA models $\zeta^1, \zeta^2, \ldots$, one for each batch. As the vocabulary of documents might change over time, AdaptivePLSA *evolves* each model, except the first one, from its predecessor model by adaption to new words.

When AdaptivePLSA evolves a PLSA model, we are in the situation depicted in Figure 3.4. A predecessor PLSA model was learned with respect to documents of the batch $\vec{D}^i$ and now should be adapted to documents of the next batch $\vec{D}^{i+1}$. The following more uncluttered notation is used in the following.

Notation:

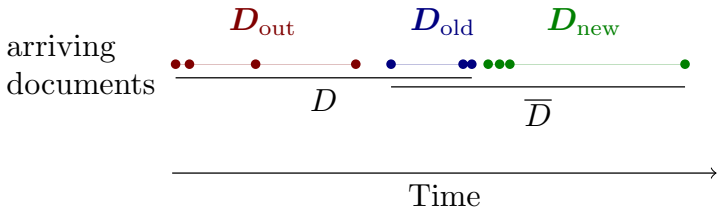


Figure 3.4: Two successive positions i and $i+1$ of sliding window covering documents D and $\overline{D}$, respectively. Also shown: annotation of corresponding partial document sets.

- batches $\vec{D}^i$ and $\vec{D}^{i+1}$ are denoted by D and $\overline{D}$
- vocabulary of D and $\overline{D}$ is denoted by W and $\overline{W}$
- d, $\overline{d}$, w, $\overline{w}$ are IDs in D, $\overline{D}$, W, $\overline{W}$
- PLSA models ζ and $\overline{\zeta}$ correspond to batches D and $\overline{D}$
- out-dated documents D_{out} are those that are in D but not in $\overline{D}$
- new documents D_{new} are those that are in $\overline{D}$ but not in D
- old documents D_{old} are those that appear in $\overline{D}$ and D
- out-dated words W_{out} are word IDs of words that appear in W but not in $\overline{W}$
- new words W_{new} are word IDs of words that appear in $\overline{W}$ but not in W
- old words W_{old} are word IDs of words that appear in W and $\overline{W}$

AdaptivePLSA evolves a PLSA model by adapting it to both: new documents and new words. This includes to incorporate new and to *forget* out-dated words. Forgetting out-dated words prevents from accumulating all words seen so far and from unnecessarily blowing up the feature space. Roughly, AdaptivePLSA goes through the following seven steps to evolve a PLSA model ζ into an updated model $\overline{\zeta}$. A visualization of these steps is presented in Figure 3.5.

Seven steps of AdaptivePLSA:

start with PLSA model ζ

1. remove topic-mixture proportions of out-dated documents D_{out}
2. fold-in new documents D_{new} into model ζ using its document-based form
3. turn model ζ into its word-based form by Bayesian calculus
4. remove topic-mixture proportions $P(z|w)$ of out-dated words W_{out}
5. fold-in new words W_{new}
6. turn intermediate PLSA model into document-based form using Bayesian calculus
7. recalibrate intermediate model by running the EM algorithm using data $\overline{D}$

result: PLSA model $\overline{\zeta}$

Each of these steps is described in depth in the next section.

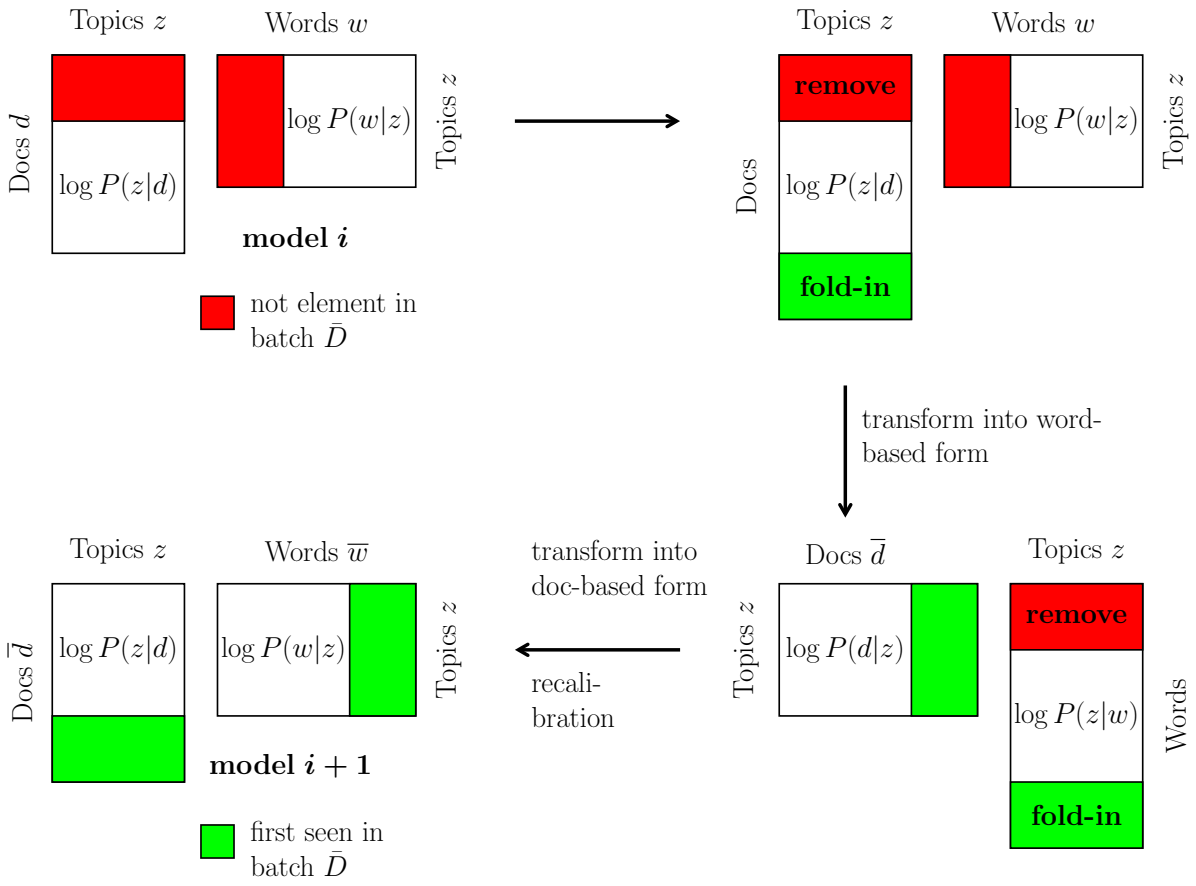


Figure 3.5: Overview of AdaptivePLSA that adapts a PLSA model i, which has been learned for documents D, to the follow-up model $i+1$ for $\bar{D}$. Logarithmic topic-mixture proportions ($\log P(z|d)$) and word-topic associations ($\log P(w|z)$) are visualized by rectangles, which correspond to their matrix representations. Document probabilities are omitted.

3.3.4 Adapting PLSA models

AdaptivePLSA starts with a learned PLSA model $\zeta = (\vec{\theta}, \vec{\omega}, \vec{\delta})$ in its document-based form. This model has logarithmic topic-mixture proportions $\vec{\theta}_d$ and document probabilities δ_d for all documents $d \in D$ and logarithmic word-topic associations for all K topics and words $w \in W$.

Remove out-dated documents

First, AdaptivePLSA removes topic-mixture proportions $\vec{\theta}_d$ of out-dated documents d with $d \in D_{\text{out}}$ by deleting $\vec{\theta}_d$ from $\vec{\theta}$. Removing these parameters is possible as they are independent of the other model parameters. When viewing the PLSA model in the proposed matrix representation (cf. Figure 3.5), all rows of the matrix $\vec{\theta}$ are independent of the other model parameters and AdaptivePLSA deletes rows that correspond to out-dated documents. To advert the temporary nature of the reduced logarithmic topic-mixture proportions, these are denoted by $\widetilde{\vec{\theta}}$. The intermediate PLSA model has parameters $\widetilde{\zeta} = (\widetilde{\vec{\theta}}, \vec{\omega}, \vec{\delta})$.

Fold-in new documents

Next, AdaptivePLSA integrates new documents D_{new} into the intermediate model. For folding-in new documents, AdaptivePLSA reduces the new documents to the vocabulary W of the current PLSA model. Then, AdaptivePLSA determines logarithmic topic-mixture proportions of the new documents by folding-in as described in Section 3.3.1. In matrix representation, folding-in new documents means to append rows, one for each new document, to the matrix of topic-mixture proportions. The result is a $|\overline{D}| \times K$ matrix $\widetilde{\vec{\theta}} := [\vec{\theta}_d; \vec{\theta}_{d'}]$ with $d \in D_{\text{old}}$ and $d' \in D_{\text{new}}$.

AdaptivePLSA resets logarithmic document probabilities $\widetilde{\vec{\delta}} = [\delta_d]$ with $d \in \overline{D}$ by new Maximum-Likelihood estimates. For these estimates it uses the documents D_{old} (not reduced) and the new documents D_{new} which have been reduced to words of the vocabulary W. The length of the vector $\widetilde{\vec{\delta}}$ is equal to the number of documents $\overline{D}$.

Turn PLSA model into word-based form

AdaptivePLSA transforms the current document-based PLSA model $\widetilde{\zeta} = (\widetilde{\vec{\theta}}, \vec{\omega}, \widetilde{\vec{\delta}})$ into its equivalent word-based form $\widetilde{\zeta}_w$. The parameters of $\widetilde{\zeta}_w$ are logarithmic topic-mixture proportions for words $\vec{\vartheta}$, logarithmic document-topic associations $\vec{\varpi}$, and logarithmic word probabilities ρ (cf. Figure 3.3, Section 3.3.2).

Logarithmic topic-mixture proportions for words could be seen as a $|W| \times K$ matrix. Each row of this matrix contains conditional log-probabilities $\log P(z = k|\omega = w)$ for topics $1 \leq k \leq K$ given the word w to which this row corresponds. Logarithmic document-topic associations could be seen as a $K \times |\overline{D}|$ matrix. The k^{th} row contains conditional log-probabilities $\log P(d|z = k)$ of document IDs given the k^{th} topic. This matrix representation is used in Figure 3.5.

AdaptivePLSA uses Bayesian calculus for turning a document-based PLSA model into its equivalent word-based form. To this end, AdaptivePLSA first computes unconditional

topic probabilities. The expression $d \in \overline{D}$ indicates a run over all document IDs of the batch $\overline{D}$. Unconditional topic probabilities are computed by

$$P(z=k) = \sum_{d \in \overline{D}} P(d)P(z=k|d) \tag{3.3}$$

$$= \sum_{d \in \overline{D}} \exp(\widetilde{\delta}_d) \cdot \exp(\widetilde{\theta}_{d,k}) \quad . \tag{3.4}$$

Next, AdaptivePLSA computes logarithmic document-topic associations $\varpi_{k,d}$, which are defined as $\varpi_{k,d} = \log P(d|z=k)$, for documents $d \in \overline{D}$ and all K topics by

$$P(d|z=k) = \frac{P(z=k|d)P(d)}{P(z=k)} \tag{3.5}$$

$$= \frac{\exp(\widetilde{\theta}_{d,k}) \cdot \exp(\widetilde{\delta}_d)}{P(z=k)} \quad . \tag{3.6}$$

Third, AdaptivePLSA computes logarithmic word probabilities $\rho_w = \log P(w)$ with $w \in W$ by marginalization as

$$P(w) = \sum_{k=1}^{K} P(w|z=k)P(z=k) \tag{3.7}$$

$$= \sum_{k=1}^{K} \exp(\omega_{k,w}) \cdot P(z=k) \quad . \tag{3.8}$$

Last, AdaptivePLSA determines logarithmic topic-mixture proportions $\vartheta_{w,k}$, which are defined as $\vartheta_{w,k} = \log P(z=k|w)$ for each word $w \in W$ and each topic by

$$P(z=k|w) = \frac{P(w|z=k)P(z=k)}{P(w)} \tag{3.9}$$

$$= \frac{\exp(\omega_{k,w}) \cdot P(z=k)}{\exp(\rho_w)} \quad . \tag{3.10}$$

At this point, the intermediate PLSA model in its word-based form is $\widetilde{\zeta}_w = (\vec{\vartheta}, \vec{\varpi}, \vec{\rho})$.

Remove out-dated words

AdaptivePLSA removes logarithmic topic-mixture proportions of out-dated words by deleting the corresponding rows of $\vec{\vartheta}$. This is possible as the topic-mixture proportions of out-dated words are independent of all other parameters of the word-based PLSA model. In matrix notation, AdaptivePLSA removes rows of the $|W| \times K$ matrix $\vec{\vartheta}$. The remaining matrix $\widetilde{\vec{\vartheta}} := [\vec{\vartheta}_w]$ has only rows that correspond to words $w \in W_{old}$. The current PLSA model reads as $\widetilde{\zeta}_w = (\widetilde{\vec{\vartheta}}, \vec{\varpi}, \vec{\rho})$.

Fold-in new words

Folding-in new words as proposed in [5] aims at incorporation of new words into an already learned PLSA model. For folding-in new words, AdaptivePLSA follows the idea of folding-in new documents. That is, it estimates topic-mixture proportions of new words by running the EM algorithm while having fixed the remaining parameters of the intermediate word-based PLSA model.

For folding-in new words, AdaptivePLSA considers all occurrences of a new word $w \in W_{new}$ in a document $d \in \overline{D}$. For each such word occurrence a pair (w, d) of a word ID and a document ID with $w \in W_{new}$ and $d \in \overline{D}$ is added to $\vec{F}$. The vector $\vec{F} = ((d,w)_1, \ldots, (d,w)_{|\vec{F}|})$ with $|\vec{F}|$ denoting the number of elements in $\vec{F}$ summarizes all these pairs.

The EM algorithm for folding-in new words works as follows. In its $(t+1)^{\text{th}}$ iteration, the EM algorithm determines posteriors for all data points of $\vec{F}$ in its E step. To this end, it uses the model parameters which have been estimated in the previous iteration.

$$1 \leq i \leq |\vec{F}|, 1 \leq k \leq K : \gamma_{i,k}^{(t+1)} = \frac{\exp(\varpi_{k,d_i}) \exp(\vartheta_{w_i,k})^{(t)}}{\sum_{\bar{k}=1}^{K} \exp(\varpi_{\bar{k},d_i}) \exp(\vartheta_{w_i,\bar{k}})^{(t)}} \tag{3.11}$$

Logarithmic document-topic associations $\vec{\varpi}$ are fixed. The topic-mixture proportions for new words $w \in W_{new}$ are re-estimated by

$$\exp(\vartheta_{w,k})^{(t+1)} \propto {}^{\alpha}/_{K} + \sum_{\substack{1 \leq i \leq |\vec{F}| \\ w_i = w}} \gamma_{i,k}^{(t+1)} \quad . \tag{3.12}$$

The EM algorithm continually runs through these E and M steps until a stopping criterion is fulfilled.

Derived logarithmic topic-mixture proportions of new words are added to the model parameters $\widetilde{\vartheta} = [\vec{\vartheta}, \vec{\vartheta}_{w'}]$ with $w' \in W_{new}$. Logarithmic word probabilities are reset by new Maximum-Likelihood estimates for all words $\overline{W}$ according to their relative frequency among all documents $\overline{D}$. The intermediate PLSA model now is $\widetilde{\zeta}_w = (\widetilde{\vartheta}, \vec{\varpi}, \widetilde{\vec{\rho}})$. In Figure 3.5, the parameters $\widetilde{\vartheta}$, and $\vec{\varpi}$ are represented by a $|\overline{W}| \times K$, and a $K \times |\overline{D}|$ matrix, respectively.

Turn PLSA model into document-based form

Next, AdaptivePLSA transforms the current word-based PLSA model $\widetilde{\zeta}_w$ into its equivalent document-based form. Topic and document-probabilities are determined by the following equations.

$$P(z=k) = \sum_{w \in \overline{W}} P(w|z) P(w) \tag{3.13}$$

$$= \sum_{w \in \overline{W}} \exp(\widetilde{\vartheta}_{w,k}) \cdot \exp(\widetilde{\rho}_w) \tag{3.14}$$

$$P(d) = \sum_{k=1}^{K} P(d|z=k)P(z=k) \tag{3.15}$$

$$= \sum_{k=1} \exp(\widetilde{\varpi}_{k,d}) \cdot P(z=k) \tag{3.16}$$

As a result, one obtains $\widetilde{\delta}_d = \log P(d)$ with $d \in \overline{D}$. For all documents $d \in \overline{D}$, AdaptivePLSA computes topic-mixture proportions by

$$P(z=k|d) = \frac{P(d|z=k)P(z=k)}{P(d)} \tag{3.17}$$

$$= \frac{\exp(\widetilde{\varpi}_{k,d}) \cdot P(z=k)}{\exp(\delta_d)} \tag{3.18}$$

and one obtains parameters $\widetilde{\theta}_{d,k} = \log P(z=k|d)$. Last, for all words $w \in \overline{W}$, AdaptivePLSA computed word-topic associations by

$$P(w|z=k) = \frac{P(z=k|w)P(w)}{P(z=k)} \tag{3.19}$$

$$= \frac{\exp(\vartheta_{k,w}) \cdot \exp(\rho_w)}{P(z=k)} \quad . \tag{3.20}$$

Logarithmic word-topic associations are $\widetilde{\omega}_{k,w} = \log P(w|z)$ with $1 \leq k \leq K$. The intermediate document-based PLSA model is $\widetilde{\zeta} = (\widetilde{\vec{\theta}}, \widetilde{\vec{\omega}}, \widetilde{\vec{\delta}})$.

Recalibrate

Finally, AdaptivePLSA jointly recalibrates parameters of the current PLSA model $\widetilde{\zeta}$. The goal of this recalibration is to exchange mutual information among the new parameters. This is necessary, as topic-mixture proportions and word-topic associations of new documents and new words have been computed separately from each other so far. AdaptivePLSA continues the EM procedure for a few iterations as described in Section 2.6.1. It uses all word occurrences of the documents $d \in \overline{D}$. After recalibration, the final PLSA model is $\overline{\zeta}$ which has been evolved from the previous PLSA model ζ by adaption to new documents and new words.

Limitations of AdaptivePLSA

AdaptivePLSA has two limitations. First, as it might happen that new documents contain only words yet unknown by the current PLSA model, some new documents might not be part of the evolved PLSA model $\bar{\zeta}$. Second, as a consequence of the first point, AdaptivePLSA might fail to incorporate some new words, for example if these only occur in documents that could not be folded-in. The risk of missing new documents and words might be lowered, e.g., by defining successive batches such that these considerably overlap each other.

3.3.5 Index-based topic threads

Applied to a stream of documents, AdaptivePLSA learns a sequence of PLSA models $\zeta^1, \zeta^2, \ldots$, each consists of K topics. These topics track the contents of the document stream over time. As each PLSA model evolves from the previous one, the k^{th} topic of a model evolves from the k^{th} topic of the previous model. The sequence of topics with the same index forms an *index-based topic thread*. Here, index refers to the value of the index variable $1 \leq z \leq K$ which enumerates the K topics in each model. In more detail, the k^{th} index-based topic thread is the sequence of word-topic associations $\exp(\omega)^1_k, \exp(\omega)^2_k, \ldots$ learned from batches $\vec{D}^1, \vec{D}^2, \ldots$. As a consequence, the k^{th} index-based topic thread describes the evolution of the k^{th} topic over time.

3.4 Baseline approach

As explained in Section 3.1, almost all methods for topic monitoring over time assume a static vocabulary over time. This assumption is questionable in context of real stream data because future vocabulary is unknown in the present. A baseline approach, which takes vocabulary evolution into account, is learning topic models independently of each other for each of the batches $\vec{D}^1, \vec{D}^2, \ldots$. The vocabulary of these batches might change over time. As each model of the sequence $\zeta^1, \zeta^2, \ldots$ is learned anew with documents of its corresponding batch, the vocabulary of the models might change over time. Thus, the models adapt to the changing vocabulary. In this work, the models that are learned independently of each other are PLSA models and this baseline approach is called *IndependentPLSA*. Inline with AdaptivePLSA, IndependentPLSA learns model parameters with the help of the EM algorithm and by maximizing the a-posteriori probability of the model parameters. It uses the same prior configurations as AdaptivePLSA does. A simple approach for extracting threads of topics from a sequence of PLSA models which have been learned by IndependentPLSA is measuring similarity between topics of successive models and connecting those by a thread that are most similar to each other.

3.5 Evaluation framework

Learning topic evolution with simultaneous adaption to new documents and an evolving vocabulary has not been intensively studied in the past. The objective of this evaluation framework is the comparison of AdaptivePLSA with IndependentPLSA. Omitting the additional background component used by Mei and Zhai [39] renders their approach for learning a sequence of PLSA models equivalent to IndependentPLSA. That background component uses knowledge of future words and, thus, cannot be applied to a real stream-scenario.

In absence of ground truth, AdaptivePLSA and IndependentPLSA are compared by determining perplexities for the learned PLSA models. Informally, perplexity measures how surprised a probabilistic model is with respect to yet unknown data. In other words, it assesses the capability of generalizing to unseen data; the less the model is surprised the better it generalizes. Hence, lower perplexities are better. In this work, two different perplexities are used: perplexity with respect to hold-out data and predictive perplexity. Both perplexities are very similar but a subtle difference exists. The hold-out perplex-

ity measures how well does the model recognize hold-out parts of training documents, whereas the predictive perplexity measures how well does the model recognize future documents.

Hold-out perplexity

For PLSA model ζ^i the hold-out perplexity is computed with respect to word occurrences in documents of the corresponding batch $\vec{D}^i$. The hold-out data of batch $\vec{D}^i$ are constructed by randomly splitting all word occurrences in documents of $\vec{D}^i$ into two disjoint parts: a training part (80%) and a hold-out part (20%).

In more detail, two counters for occurrences of word w in document d of batch $\vec{D}^i$ are maintained: $a_{d,w}$ for the training part and $b_{d,w}$ for the hold-out part. Both counters sum up to the total number of occurrences of word w in document d and $1/4 a_{d,w} \approx b_{d,w}$. The hold-out counts are stored in the vector $\vec{B}$.

PLSA models are learned using the training data only, i.e., all word occurrences of word in documents of batch $\vec{D}^i$ as described by the training counts. The hold-out perplexity for a learned model ζ^i with respect to the hold-out data of batch $\vec{D}^i$ reads as

$$\text{perplex}(\zeta^i) \quad = \quad \exp\left(-\frac{\sum\limits_{b_{d,w}\in\vec{B}} b_{d,w}\log\left[\sum_{k=1}^{K}\exp(\theta_{d,k})\cdot\exp(\omega_{k,w})\right]}{\sum\limits_{b_{d,w}\in\vec{B}} b_{d,w}}\right) \tag{3.21}$$

The topic-mixture proportions and word-topic associations are parameters of the model ζ^i.

Predictive perplexity

The predictive perplexity for PLSA model ζ^i is measured with respect to future documents of the next batch $\vec{D}^{i+1}$. Future documents must be folded-in into the PLSA model to determine their topic-mixture proportions which are necessary for computing the predictive perplexity [18, 39]. In this work half-folding-in as suggested by Welling et al. [52] is used as this approach is supposed to give more realistic results.

First, words from the future documents in $\vec{D}^{i+1}$ are removed that do not belong to the vocabulary known by the PLSA model ζ^i. Then, the reduced future documents are split into a fold-in (50%) and a hold-out part (50%). As in the case of hold-out perplexity, two counters of occurrences of word w in each document d in $\vec{D}^{i+1}$ are maintained: $a_{d,w}$ and $b_{d,w}$ with $a_{d,w} \approx b_{d,w}$. Folding-in and hold-out counts are stored in the vectors $\vec{A}$ and $\vec{B}$, respectively.

Future documents are folded-in into the PLSA model using folding-in data as described by counts $\vec{A}$. Folding-in extends the model ζ^i by estimates of logarithmic topic-mixture proportions of the future documents. Afterwards, the predictive perplexity for the extended PLSA model ζ^i with respect to the hold-out data of the future documents $\vec{B}$ is computed with Equation 3.21. Computing the predictive perplexity is possible for all models of a sequence except for the last model as for the last model no future documents are available, yet.

3.5.1 Influence of hyper-parameters

The choice of hyper-parameters might have an considerable effect on the perplexity of topic models as discussed in Section 2.6.3. To analyze this effect, PLSA models with different setting of hyper-parameters α and β (see Figure 2.3, p. 12) are learned and the hold-out perplexities are determined.

3.5.2 Influence of learning procedure

For a sequence of document batches $\vec{D}^1, \vec{D}^2, \ldots$ AdaptivePLSA and IndependentPLSA learn a sequence of PLSA models $\zeta^1, \zeta^2, \ldots$. Parameter learning of these models by AdaptivePLSA and IndependentPLSA mainly differs by initialization of the model parameters. IndependentPLSA uses random initialization whereas AdaptivePLSA uses learned parameters of a previous model as start configuration of the next model. To investigate whether this coupling indeed propagates useful information for parameter learning of the next model, hold-out perplexities of models learned with AdaptivePLSA and IndependentPLSA are compared.

In addition, the effect of the number of learning iterations m on the hold-out perplexity is considered. In more detail, the number of learning iterations is either the number of EM iterations used by IndependentPLSA or the number of EM iterations used for model recalibration by AdaptivePLSA. To make this comparison fair, the EM algorithm used by IndependentPLSA is restarted three times. Only those parameters that give the best a-posteriori probability are used for model parametrization.

3.5.3 Influence of natural stream order

The impact of the natural order of the streaming documents on the performance of predicting future documents is investigated. To this end, the streaming documents are considered in their original (natural) order and in a permuted order. AdaptivePLSA and IndependentPLSA are used for learning sequences of PLSA models from the natural and permuted document stream. Afterwards, predictive perplexities are computed. If the order of streaming documents contains useful latent information, then the predictive perplexities computed for the permuted stream should be larger than those obtained for the natural stream.

3.5.4 Meaningfulness of index-based topic threads

Index-based topic threads are primarily defined via a technical parameter, i.e., the value of the index variable $1 \leq z \leq K$. This index variable simply enumerates the K topics of a PLSA model. This experiment sheds light on how effective this construction of index-based topic threads is. To this end, the semantic meaningfulness of index-based topic threads is investigated.

Each index-based topic thread defines index-based pairs of successive topics, i.e., pairs of topics with the same index in two successive models. In contrast thereto, best-matching pairs of topics from two successive models are pairs of topics that are most similar to each other. As best-matching pairs of topics are defined with respect to their content, i.e., with respect to their word-topic associations, best-matching pairs better agree with our intuition about meaningful threads of topics. Consequently, the

Year	2000	2001	2002	2003
$\|\boldsymbol{D}\|$	57	71	85	87

Year	2004	2005	2006	2007
$\|\boldsymbol{D}\|$	115	121	133	194

Table 3.1: Number of documents per year for the SIGIR data set.

semantic meaningfulness of index-based topic threads is assessed by how many of their index-based pairs agree with the definition of best-matching pairs.

For the purpose of determining best-matching pairs, it is necessary to define a similarity measure between pairs of topics. In this experiment, the similarity between two topics is measured by the cosine similarity between their word-topic associations. These associations are vectors in $\mathbb{R}^{|W^i|}$ and $\mathbb{R}^{|W^{i+1}|}$ with $|W^i|$ denoting the size of vocabulary of batch $\vec{D}^i$. As both sizes are very likely different both vectors are embedded into the space spanned by the joint vocabulary of both batches. Missing word-topic associations are filled with zero-probabilities. From a sequence of PLSA models $\zeta^1, \zeta^2, \ldots, \zeta^{\bar{N}}$ the set of all best-matching pairs is determined as follows. First, the most similar follow-up topic is determined for each topic of the models up to $\zeta^{\bar{N}-1}$. Then all pairs are neglected whose cosine similarity is lower than a given threshold MinSim. Omitting best-matching pairs with a very low similarity is necessary in order to keep only meaningful best-matching pairs with a reasonable high similarity.

The percentage of index-based pairs that are also best-matching pairs is reported. If topologies of topic-based index threads and best-matching pairs agree well (the reported percentage is high), then the index-based topic threads are assumed to be indeed meaningful.

3.6 Experiments

In this section, details about the data set are given and the results are discussed.

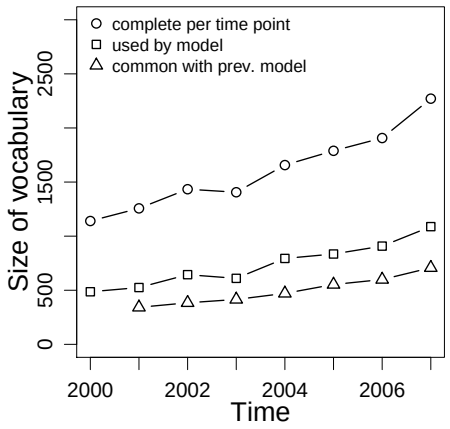
3.6.1 Data set

Articles of the ACM Digital Library which have been published at the ACM SIGIR[2] conference between 2000 and 2007 were used. The titles and abstracts of these articles are easily accessible through the ACM Digital Library. Posters were excluded because they often do not have abstracts. A document was defined for each remaining article by merging its title and abstract. Last, English stopwords were removed and the Porter stemmer was applied to the documents. This data set is called SIGIR data hereafter.

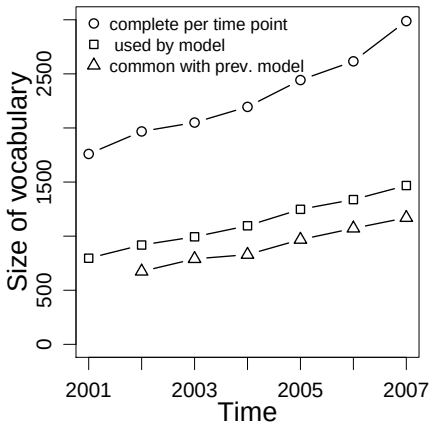
The number of SIGIR documents of the constructed data set varies per year as shown in Table 3.1. Their number increases from 2000 to 2007. The reason might be an increasing popularity of the SIGIR conference.

A stream of SIGIR documents was defined by ordering the SIGIR documents according to their year of publication. Sliding windows of size one year and two years were

[2]Association for Computing Machinery Special Interest Group in Information Retrieval `www.sigir.org`, May 3, 2012



(a) Sliding window covers all documents per year ($t = 1$) and shifts by one year.



(b) Sliding window covers all documents of two successive years ($t = 2$) and shifts by one year.

Figure 3.6: Word statistics of SIGIR data for different lengths of the sliding window. Complete vocabulary is the vocabulary before stopword removal, stemming and learning.

applied to this document stream. Regardless of their size, the windows were shifted by one year. This resulted in 8 and 7 batches of size one and two years, respectively. Words that appeared only in one document per batch were removed from the vocabulary of the corresponding batch. This reduced the vocabulary/feature space but would not harm the learning of topics as these are patterns of frequently co-occurring words.

Word statistics of the SIGIR documents are presented in Figure 3.6. The overall vocabulary increases from 2000 to 2007 by a factor of about 2. A reason for this observation could be the increasing number of published documents. The vocabularies utilized by the PLSA models increase by about the same factor, although their sizes are smaller than the total vocabulary of the corresponding batch. Three reasons could be (i) omitting of rare words, (ii) removing stopwords, and (iii) it might have happened that some words could not be incorporated into the model as described in Section 3.3.4 (Limitations of AdaptivePLSA). Especially, removing rare words might reduce the size of the vocabulary substantially as it is known that word frequencies often follow the Zipf's law which says, simplistically speaking, that only a few words occur occur very frequently whereas a lot of words occur rarely. Last, successive models have at least 50% of their vocabulary in common. This percentage is, on average, higher for batches of size two years than for batches of size one year. This agrees with the expectation that partially overlapping batches should lead to vocabularies that overlap to a higher degree.

3.6.2 Impact of hyper-parameters

Hyper-parameters α and β are the parameters of the Dirichlet priors for topic-mixture proportions and topic-word associations, respectively (see Section 2.6.1). These hyper-parameters were varied such that the real pseudo counts ($\alpha/K, \beta/KM$) of MAP parameter estimates are in $\{0.01, 0.1, 1, 10, 100\}$.

For this experiment, one single batch that contained all documents from 2000 to 2007

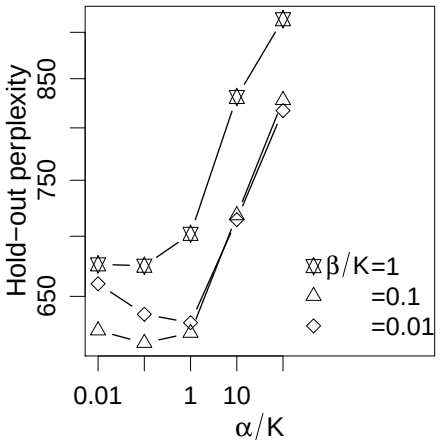


Figure 3.7: Influence of priors on hold-out perplexity (lower perplexities are better). Hyper-parameters α and β control the Dirichlet priors for topic-mixture proportions and word-topic associations, respectively. Results for the combination $^{\alpha}/_{K} \in \{10, 100\}$ are not shown because they lead to worse perplexities.

was used and the number of topics was fixed at $K = 10$. Word occurrences were randomly partitioned into 80% training data and 20% hold-out data. Hold-out perplexities were determined with respect to the hold-out data after PLSA models have been learned with different setting of the hyper-parameters. Learning was repeated 50 times and the obtained average hold-out perplexities are shown in Figure 3.7.

The hyper-parameters indeed affect the perplexities. The combination $^{\beta}/_{KM} = 0.1$ and $^{\alpha}/_{K} = 0.1$ gives best hold-out perplexity. These results indicate that

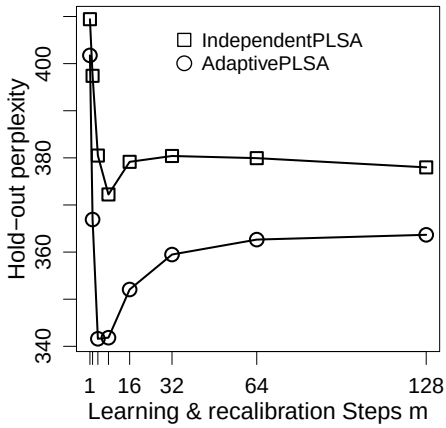
1. the Maximum-A-Posteriori principle of parameter estimation opens room for improving the generalization capability of PLSA models as optimal hyper-parameters might improve the hold-out perplexity

2. optimizing hyper-parameters is valuable as this might lead to better parameter estimates in terms of hold-out perplexity

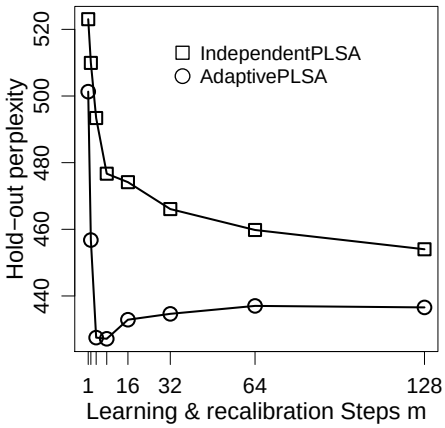
The determined optimal setting of hyper-parameters was used in the following experiment.

3.6.3 Influence of learning procedure

The two learning procedures AdaptivePLSA and IndependentPLSA were used in this experiment and the number of learning iterations $m \in \{1, 2, 4, 8, 16, 32, 64, 128\}$ was varied. The number of topics was fixed at $K = 10$. In general, a trained PLSA model will fit the training data better when the number of learning iterations increases. On the other hand, the better a PLSA model is fit to the training data the worse it may generalize from these data and predict yet unseen hold-out data. PLSA models that well generalize to hold-out data better model the overall characteristics of the streaming documents. A number of 50 sequences of PLSA models were learned for each combination of a learning procedure, a length of the sliding window (one and two years), and a number of learning iterations m. Hold-out perplexities that have been averaged over all models of all 50 repeats per combination are depicted in Figure 3.8.



(a) Sliding window captures all document per year $t = 1$ and slides by one year.



(b) Sliding window covers all documents of two successive years $t = 2$ and shifts by one year.

Figure 3.8: AdaptivePLSA vs. IndependentPLSA and impact of number of learning EM steps m on average hold-out perplexity.

Figure 3.8(a) reveals that AdaptivePLSA is generally superior over IndependentPLSA as hold-out perplexities for AdaptivePLSA are smaller than the corresponding ones for IndependentPLSA. This tendency is independent of the number of EM steps m. Moreover, hold-out perplexities have a local minimum for both learning approaches. The number of EM steps needed to reach this minimum are those for which one obtains optimally generalizing PLSA model. Interestingly, the numbers of optimal EM steps are small: 4 EM steps for AdaptivePLSA and 8 EM steps for IndependentPLSA. A reason for this finding could be that titles and abstracts are very condensed descriptions; they use very conscious and specific words. Fitting a PLSA model too close to the specific words of training abstracts and titles, one might quickly end up with PLSA models that poorly generalize to hold-out data. Further on, the number of EM iterations needed to give optimal PLSA models is slightly smaller in the case of AdaptivePLSA compared to IndependentPLSA. In summary, using AdaptivePLSA leads to better hold-out perplexities in less EM steps.

Figure 3.8(b) reveals almost the same tendencies with respect to the comparison of AdaptivePLSA and IndependentPLSA. One difference is that the hold-out perplexities for IndependentPLSA show now local minimum; hold-out perplexities steadily decrease toward $m = 128$ learning iterations. The overall best hold-out perplexity is obtained when using AdaptivePLSA and $m = 4$ learning steps. All hold-out perplexities of PLSA models learned with IndependentPLSA are substantially higher (worse). As AdaptivePLSA needs less learning iterations and still learns models that better generalize to hold-out data, AdaptivePLSA turns out to be superior over IndependentPLSA.

3.6.4 Impact of natural stream order

AdaptivePLSA and IndependentPLSA were used to learn 50 sequences of PLSA models from the SIGIR document stream in its natural stream order. Learning was repeated a

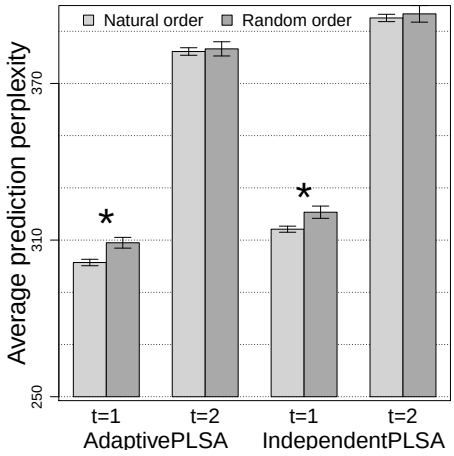


Figure 3.9: Impact of natural stream order on capability of predicting new documents. Again, two different sizes of the sliding window are taken into account (one $t = 1$ and two $t = 2$ years). The star notes significant differences according to a t-test with significance level 0.05.

second time having permuted the time stamps of the documents each turn anew. The sizes of the new batches were equal to the sizes of the original batches. Again, the number of topics was set equal to $K = 10$.

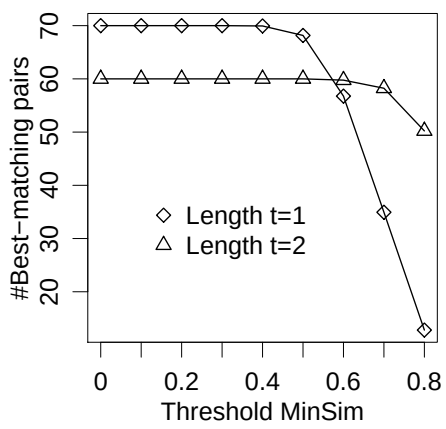
Averaged predictive perplexities per model for each combination of a learning procedure and stream order are presented in Figure 3.9. The natural stream order leads to significantly better predictive perplexities if the sliding window is of size one year. This result is true for both learning approaches. Figure 3.9 indicates further that AdaptivePLSA benefits slightly more from the natural stream order than IndependentPLSA. If the sliding window is of size two years, then the impact of the natural stream order is less obvious. The reported predictive perplexities are almost the same for the natural and the permuted order. Two reasons could be (i) the larger size of training documents as each PLSA model was learned with documents of two successive years, and (ii) a larger vocabulary of the training data. A larger number of training documents could lead to better parameter estimates and so might enhance the predictive perplexity. A larger vocabulary might include future words with a higher probability and so might lead to PLSA models that are capable of better predicting future documents.

Last, adaptively learned PLSA models lead to lower predictive perplexities than independently learned PLSA models. This finding is in agreement with previous results and indicates that AdaptivePLSA has the potential to learn PLSA models that better generalize to yet unseen data.

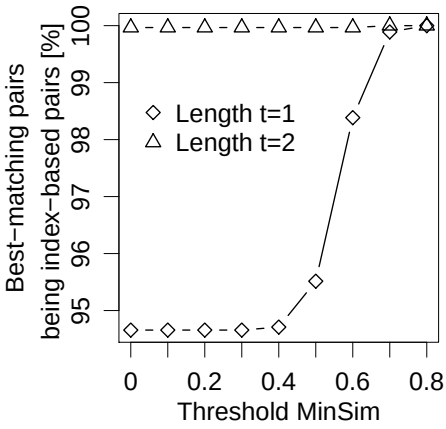
3.6.5 Meaningfulness of index-based topic threads

A sequence of PLSA models with $K = 10$ topics was learned from the SIGIR document stream in natural order with AdaptivePLSA. The meaningfulness of index-based topic threads was measured by opposing them to best-matching pairs.

Figure 3.10(a) shows that the largest number of best-matching pairs is obtained if the lower threshold MinSim on the cosine similarity of best-matching pairs is equal to zero (MinSim = 0.0). This means that each topic has a best-matching follow-up topic



(a) Total number of best-matching pairs. MinSim is the lower bound of similarity between consecutive topics.



(b) Percentage of best-matchings that connect successive topics having the same topic index.

Figure 3.10: Best-matching pairs vs. index-based topic threads. Length of sliding window: one ($t = 1$) and two ($t = 2$) years.

as all best-matching pairs are considered. The largest possible number of best-matching pairs is 70 and 60 for a sliding window of size one and two years, respectively. For a sliding window of size one year this number is 70 because each model has $K = 10$ topics and the number of transitions between two successive batches is seven. Using a sliding window of size two years, the number of transition is six and so the maximal number of best-matching pairs is 60. As the threshold MinSim increases to 0.8, the number of best-matching pairs decreases below 20 and 50 for a window size of one and two years, respectively. As all remaining best-matching pairs (see Figure 3.10(b)) are also index-based pairs, these best-matching pairs connect only topics with the same index. Increasing the threshold further would leave the percentage of best-matching pairs that are also index-based pairs unchanged at 100%. Consequently, the threshold MinSim is meaningfully varied only in the interval $[0; 0.8]$.

Figure 3.10(b) reveals that, for a window size of two years, all best-matching pairs, regardless of the value of the threshold, are index-based pairs. For a window size of one year at least 94.5% of the best-matching pairs agree with index-based pairs. This percentage increases to 100% when the threshold MinSim reaches 0.8. This finding indicates that strong best-matching pairs, which are those that remain after increasing the threshold MinSim to 0.8, tend to connect only topics with the same index. This demonstrates that AdaptivePLSA learns sequences of PLSA models such that topics with the same topic index in successive models could be meaningfully connected. The resulting index-based topic threads agree with best-matching pairs to a high degree. As such, index-based topic threads are similar to those topic threads proposed by Mei and Zhai [39] that are constructed by matching topics with respect to their similarity.

Figure 3.11: Index-based topic threads for SIGIR data from 2000 to 2007. For each topic the top-25 most likely words are shown.

K	2000	2001	2002	2003	2004	2005	2006	2007
5	user retriev search	model queri user	model inform re-	queri inform model	model queri retriev	model queri re-	model queri re-	queri model re-
	model inform ap-	translat retriev	triev user queri	retriev user imag	inform languag	triev inform term	triev inform term	triev term inform
	proach process	system inform	translat techniqu	word languag tech-	user imag term	approach imag lan-	languag user base	languag approach
	queri cognit tech-	search result im-	languag term	niqu automat show	translat present	guag user translat	system probabl	improv base word
	niqu need system	prov provid list	statist base system	gener interact base	automat word	word annot gener	result approach	imag gener tech-
	structur translat	imag studi paper	improv present	effect annot term	show approach	effect propos paper	gener translat	niqu context show
	base studi propos	estim feedback	word resourc inter-	studi translat de-	content effect	probabilist studi	probabilist im-	user feedback sys-
	on internet gener	concept transform	fac crosslanguag	scrib music paper	clir improv ac-	problem automat	prov effect expans	tem propos paper
	problem iden-	applic approach	chines paper dic-	probabilist statist	cess provid studi	present improv	interact perform	depend perform
	tifi paper poster	resourc problem	tionari studi music	visual	resourc manual	pattern expans	context studi	sentenc suggest
	characterist	exist process	select show		propos paper	statist	paper brows word	expans
4	inform document	topic web page	web topic extract	search web topic	web search topic	search web page	search web user	search web user
	system web retriev	document summari	search system	document system	system page doc-	user engin result	task page inform	result inform page
	qualiti topic im-	link stori detect	analysi set item	find task base relev	ument evalu task	topic system re-	result topic link	engin topic task
	prov result present	system strategi	document track	link analysi con-	result engin de-	lationship detect	sentenc algorithm	studi present
	filter combin per-	search rank inform	page engin rank	cept research set	tect analysi user	paper link task	engin differ time	queri find log
	form metric event	text extract initi	develop tool requir	inform page evalu	relev gener describ	data algorithm	studi present gener	link develop con-
	contribut effect	set import evalu	differ find data	engin specif tool	person research	summari evalu	evalu rank find	tent provid time
	propos type collect	algorithm cite au-	base evalu present	result construct	commun level	event implicit new	interest on target	gener relev system
	statist similar	tomat paper event	design text filter	human network	name link event	track locat analysi	commun social	analysi explor
	differ task relev	analysi		present	discuss extract	relev inform		interact
3	document clus-	queri languag doc-	cluster queri docu-	ir data document	document clus-	document inform	document structur	document index
	ter queri word	ument perform dis-	ment model search	question cluster	ter structur data	data question	cluster question	cluster answer
	perform inform	tribut search engin	data perform	algorithm databas	inform ir result	semant techniqu	semant space net-	method structur
	system retriev	answer data im-	retriev languag	queri answer struc-	effect techniqu	algorithm struc-	work index system	similar effici ques-
	featur result select	prov inform ques-	collect result in-	tur inform xml	semant answer	tur answer index	inform answer	tion inform propos
	collect question	tion system find	form ir method	research perform	xml index search	problem titl sim-	techniqu propos	problem applic
	distribut databas	similar result show	invert problem	combin result	databas method lsi	ilar latent applic	relat data analysi	optim base retriev
	answer find improv	task combin collect	similar file merg	search model	high paper rank	ir base method	process similar	time combin show
	text page compar	cluster investig re-	propos experi	distribut system	question approach	relat propos queri	problem effici ap-	represent space pa-
	local effect increas	triev describ ap-	time effici compar	present import	problem phrase	process effici studi	plic type complex	per local number
	test	proach	paramet	type relationship	list	show	latent repres	text
				approach				
2	evalu method	relev document	document retriev	retriev evalu doc-	retriev document	retriev perform	document relev	rank relev evalu
	retriev document	evalu retriev ef-	evalu relev result	ument perform	collect perform	relev collect result	evalu rank retriev	system collect
	relev measur precis	fect term index	index effect term	system method	evalu relev feed-	document feedback	method measur	document retriev
	base test effect ex-	method threshold	score collect test	term effect relev	back system im-	evalu measur test	collect result pre-	perform measur
	peri term approach	base highli score	trec two averag	collect measur	prov test method	experi set trec	cis set perform	method effect re-
	queri ir novel	comput measur	inform arab size	test result pro-	measur effect rank	precis function	feedback estim im-	sult test precis set
	averag larg differ	inform rank result	corpu stem precis	pos rank video	set inform result	effect term rank	prov averag show	ir judgment differ
	word propos paper	system feedback	improv measur	object approach	experi term pre-	show estim system	test experi effect	score compar topic
	filter rate result	better weight pro-	comput paper	similar expans av-	cis show blind	distribut improv	sampl trec element	two propos estim
		pos paper differ	show	erag probabl trec	function length	method inform	on metric	trec
		techniqu		improv investig	video			
1	classifi text	text method ap-	text algorithm	text classif classifi	algorithm filter	method text featur	text approach	featur algorithm
	method algorithm	proach summar	method document	featur method	featur method text	classif approach al-	method featur	learn data text
	learn classif vec-	classif segment	classifi classif	learn approach	classif user learn	gorithm show learn	algorithm data	method approach
	tor perform train	model algorithm	gener learn data	field algorithm	model perform col-	base propos train	classif base learn	classif propos
	web model docu-	gener featur per-	approach perform	gener categor seg-	labor train propos	summar extract	object rate show	problem paper
	ment predict data	form document	categori present	ment paper machin	support approach	outperform com-	propos paper train	base classifi de-
	probabilist filter	categor classifi	categor summar	vector extract re-	classifi compar pa-	par two problem	summar set filter	tect perform label
	categor system	sentenc summari	sentenc model	sult show group	per base problem	classifi data select	similar vector clas-	multipl train filter
	machin summar	train differ show	applic differ set	sentenc analysi	rate machin provid	improv label music	sifi combin compar	segment svm func-
	set hierarch space	space vector com-	predict result train	hierarchi train	improv gener	filter collabor	larg framework	tion predict task
	approach poster	bin base automat	corpu term	categori exampl				show
		term						

3.6.6 Example index-based topic threads

A sequence of eight PLSA models with $K = 5$ topics using AdaptivePLSA was learned for this example analysis. The sliding window was of size one year and it was shifted by one year. As a result, each batch covered documents of one year and batches did not overlap. Table 3.11 presents the learned topics which are arranged such that each index-based topic thread is shown along one row. The top-25 most likely words per topic are listed. The index-based topic threads presented in Table 3.11 might be interpreted as follows.

Thread 5: two main aspects: multilingual IR and natural language processing; further aspects are relevance feedback for multilingual IR, natural language processing with sub-area automatic translation and multimedia in context of presentation; another sub-area seems to be image annotation

Thread 4: main thematic subject is web; the two sub-areas on information extraction and link traversal are present at the beginning; web search and user queries appear later; other subjects are improving document rankings by exploiting citations, social networks and evaluation

Thread 3: document clustering in information retrieval is main subject of the third topic thread; data mining in databases appears in 2000 and 2003; a sub-area on semantic techniques is present from 2004 on

Thread 2: main subject seems to be evaluation of methods for information retrieval with respect to documents; a sub-area on ranking of video data seems to be present in 2003 and 2004; a sub-area on relevance feedback appears later in 2005 and 2006

Thread 1: supervised machine learning and classification; later, more elaborate aspects such as feature selection, collaborative filtering and support vector machines appear

Some index-based topic threads are relatively stable whereas others considerably change. For example, the third topic thread is clearly on document clustering and this main subject seems to be relatively stable from 2000 to 2007. In contrast, the first topic thread, which is about supervised machine learning (a multi-term that does not appear among the words), is changing considerably. This thread captures aspects of classification at the beginning. More specific aspects, e.g., on support vector machines and feature selection, appear later.

All together, these five index-based topic threads are a comprehensible summary of the thematic subjects that are present in the SIGIR document stream. As such they provide the reader with an overview of the research topics presented at the SIGIR conference from 2000 until 2007.

3.7 Conclusions and further directions

This chapter deals with learning a summary of contents of document streams, in which the thematic subjects and the vocabulary of the documents change with time. The proposed approach, which is called AdaptivePLSA, is an online learning procedure for

learning PLSA models from a document stream. By adapting PLSA models to new documents and words while removing out-dated words, AdaptivePLSA keeps the feature space up-to-date. By that, AdaptivePLSA waives the assumption that the complete vocabulary is known in advance and it alleviates the negative impact of large, continually growing vocabularies with many words of only temporary importance. The topics of the learned PLSA models reflect the contents of a given document stream. By studying and interpreting these topics, the reader is able to discover thematic subjects, which are present in the streaming documents, and their evolution over time.

AdaptivePLSA is evaluated by applying it to a stream of documents of the ACM SIGIR conference from 2000 to 2007. The hold-out perplexities of the learned PLSA models differ when the hyper-parameters of the PLSA priors are varied. This indicates that determining optimal hyper-parameters is necessary for learning PLSA models that well generalize to yet unseen data. This finding is inline with Asuncion et al. [32] who find that choosing optimal priors of topic models has an important influence on the perplexity. In addition, the number of iterations of the EM algorithm for parameter learning strongly influences perplexities. Surprisingly, a few iterations are necessary for learning PLSA models that reach best perplexities. This indicates that running the EM algorithm until convergence is unnecessary for obtaining parameter estimates which give best perplexities. An additional advantage of stopping the EM algorithm already after a few steps is that the computational costs of learning are reduced. A comparison of AdaptivePLSA with IndependentPLSA shows that AdaptivePLSA leads to PLSA models that reach better perplexities. This demonstrates the effectiveness of AdaptivePLSA in utilizing latent information from previous models for learning PLSA models from a stream of documents. Next, the intrinsic value of the order of the streaming documents was analyzed. The results show that PLSA models which have been learned by AdaptivePLSA from the SIGIR stream in its natural order better predict future documents than models learned from a permuted SIGIR stream. This finding indicates that the natural order of streaming SIGIR documents has some intrinsic value for adaptively learning PLSA models. In addition, this finding demonstrates that AdaptivePLSA is capable of exploiting this intrinsic value. Last, index-based topic threads, which are extracted from a sequence of PLSA models learned by AdaptivePLSA, strongly agree to threads of most similar topics. Thus, index-based topic threads are meaningful in the sense that they, although being defined via a mere technical parameter, agree with humans' intuition of threads of similar topics.

As streams of documents become an emerging part of modern life, they are a reasonable subject of current and on-going research of machine learning. A future direction of online learning of topic models is recurring topics. Recurring topics, when identified in a stream of documents, might be exploited for learning future topic models from the latest documents of this stream. Likewise, identifying and modeling correlations among topics that occur in a document stream at different points in time also might be exploited for enhancing learning of future topic models. Another potential direction of research is designing topic models for modeling streams of other kinds of data, e.g., audio or video data. The focus of this research would be the various data streams that people nowadays are confronted with. Examples include video streams and audio streams like online music streams. Last, machine learning approaches for learning topics from streams of documents for the inspection by a user are only as powerful as the presentation of the learned topics is. Thus, another direction of research is the development of sophisti-

cated visualization techniques for presenting topics learned from streaming documents and their evolution over time.

Chapter 4

Visually summarizing document streams

Nowadays, streams of documents are present in everyday life. Examples are annual conference proceedings, articles in newspapers or news reports, received e-mails, published research articles in scientific journals, received short messages via SMS, entries in online blogs, Twitter messages and many more. Streams of documents are a source of knowledge. For instance, documents published in the annual conference proceedings of the SIGIR conference reflect what the main research topics have been and how they changed over time. Such pieces of information are helpful for organizers who want to identify current subjects of research and how these evolved from former subjects. Another example are news articles; analyzing them over time, journalists might investigate periods of time in which a certain political debate of interest was going on.

An electronic collection of streaming documents proliferates and grows continually. Its contents change with time and so do the thematic subjects which the documents are about. If one wants to know the changing contents, one might inspect the streaming documents. But, inspecting whole documents is time consuming and, if the stream is fast, impractical. In this chapter, a visualization technique, which is meant as a tool for generating a comprehensible summary of the changing contents present in a document stream, is proposed.

Topic evolution is a relatively new research subject which encompasses the unsupervised discovery of thematic subjects in a stream of documents *and* the adaptation of these subjects as new documents arrive. Research on summarizing document collections [53–55] aims at helping readers to keep up with changing contents in growing collections. While many powerful methods for analyzing and modeling topic evolution exist, the combination of learning *and* visualization of the evolving topics has been less explored, although effective visualization is indispensable for readers who want to discover and track thematic subjects in streams of documents.

Learning topics is successfully pursued with probabilistic topic models like PLSA [18] or LDA [15] which derive a small number K of groups of words that appear frequently together in documents. These groups of words are probabilistically represented as distributions over the vocabulary. Often, only a few very likely words are presented to the readers for inspection and interpretation. In most cases, humans can interpret these few words as thematic subjects.

In the stream scenario, interpreting topics becomes more tedious as topics change

	1881	1890	1900	1910
Atomic Physics	force energy motion differ light measure magnet direct matter result	motion force magnet energy measure differ direct line result light	magnet electric measure force theory system motion line point differ	force magnet theory electric atom system measure line energy body
Neuroscience	brain movement action right eye hand left muscle nerve sound	movement eye right hand brain left action muscle sound experiment	brain eye movement right left hand nerve vision sound muscle	movement brain sound nerve active muscle left eye right nervous

Figure 4.1: Box scheme for two topics (rows) over time (columns). Example topics were inferred from the Science corpus and are taken from the manuscript on the Dynamic Topic Model [40].

continually when new batches of documents arrive. Consequently, for each batch and each of the K topics, the reader would have to study a list of words. The total number of word lists would be equal to the number of topics K times the number of batches. To ease that task a new visualization method, which is called *TopicTable* is introduced in this Chapter. TopicTable visualizes learned topics in combination with additional pieces of information useful for deducing the thematic subjects and their evolution.

TopicTable solves the following design challenges: (i) it uses the canvas efficiently, and (ii) it displays important information while retaining less important ones. To this end, the box scheme is extended that is often used for presenting topics in the literature on topic modeling. An example of the box scheme is shown in Figure 4.1; each box corresponds to one topic and lists a small number of its most likely words. Columns correspond to time periods as given by the time stamps of documents of the corresponding batches. Vertically aligned boxes present topics learned from the same batch of documents. Horizontally aligned boxes indicate relatedness among the displayed topics and emphasizes the perception of their evolution over time.

Beside the most likely words of each topic, the box scheme often neglects other pieces of information, which could be helpful for deducing thematic subjects and their evolution. TopicTable visualizes, at a glance and in an intuitive manner, the following four additional pieces of information.

1. emergence of new words

2. relative strength of topics

3. similarity of topics presented in any two successive boxes along one row

4. similarity among all topics across the whole table

All these pieces of information have the potential of assisting the reader in identifying and assessing the contents of a document stream. For example, the strength of topics might be particularly helpful for readers if they are interested in the relative prevalence of topics, or if readers are mainly interested in studying the most prevalent topics of each batch.

As lists of words and similarities among topics are the input, TopicTable might be combined with any topic model that is capable of learning topics and their evolution from document streams. In this work, AdaptivePLSA, an extension of PLSA for dynamic topic modeling discussed in Chapter 3, is used for learning topics from streams of documents. AdaptivePLSA is especially suited as learned topics with the same index over time might be meaningfully connected to an index-based topic thread as described in Section 3.3.5. Hence, topics of each index-based topic thread could be well aligned along one row of TopicTable. There is no extra need for matching successive topics to each other in order to obtain well interpretable threads of topics. The possibility to combine TopicTable with any suited topic model makes TopicTable a promising visualization tool for presenting topics and their evolution.

The remainder of this chapter is structured as follows. Related work is discussed in Section 4.1. In section 4.2, the notation for document streams is shortly recalled and *document prototypes*, an interface for coupling TopicTable with different methods for learning topics from document streams, are introduced. TopicTable is described in detail in Section 4.3 and, afterwards, the effects of parameter settings on TopicTable are discussed in Section 4.4. Afterwards, two case studies for the summarization of the contents of NIPS and SIGIR articles and their evolution over time are presented in Section 4.5. The conclusions are given in Section 4.6.

4.1 Related work

Visually summarizing contents of document streams is a way of knowledge discovery; it helps in uncovering the multiple thematic subjects of the streaming documents. Topic models especially extended for learning topics from streams of documents are well suited for exploratory analyses of the contents of a stream of documents. Beside Adaptive-PLSA [5], which extends PLSA [18] to streaming documents and which is used in this thesis, other approaches for dynamic topic modeling have been developed. Examples include [39–41, 49, 50, 56, 57]. All these approaches have in common that they model a topic as a distribution over the vocabulary, and that they adapt topics to contents of newly arriving documents. Some approaches allow for words to become obsolete and irrelevant while new words emerge [5, 49, 50]. Taking into account the evolution of the terminology, i.e., the change of vocabulary, is indispensable for learning topics and their evolution. A reason is that the evolution of thematic subjects is inevitably associated with the increased importance of some words that were irrelevant or unknown in the past [5].

Topic models [15, 18] are often accompanied by some simple visualization aids. For example, Blei et al. [40] list the most likely words for a topic at each point in time, and further visualize how the probability of a certain word being associated to a topic changes over time. Contrary to listing most likely words for a topic, Mei et al. [58] and Boyd-Graber et al. [26] point out that human inspection is facilitated by choosing the

most descriptive words for a topic rather than the *most likely* ones. Being based on *document prototypes*, TopicTable (i) might present meaningful word lists of any type, for example the most likely, the most descriptive, or the most discriminative words for each topic, and TopicTable (ii) might be coupled with different approaches for learning word patterns reflecting thematic subjects of streams of documents.

TimeFall [53] has been designed for the visualization of clusters in evolving social networks. In contrast to TopicTable, which visualizes topics learned with topic models, the clusters which are presented by TimeFall are actually *communities of words* visualized as boxes. The authors describe an example visualization as follows (cf. [53], p. 1).

> "In Figure 1 each box represents a community of words – a user profile topic. Each topic is thus described by a set of keywords that are characteristic for the corresponding topic. Each line (horizontal group of topics) represents a time step, and arrows between the topics from the adjacent time steps represent the evolution of the topics. Notice the splits and merges of the topics over time."

Beside the characteristic keywords, the size and weight of each topic are printed inside the corresponding box.

TextPool, proposed as help for decision makers, provides a summary of streaming documents [59]. Applied to a stream of documents, TextPool updates a summary for the latest resources by clustering related terms. Although TextPool deals with streams of documents, it aims at giving an overview of the most recent contents. In contrast thereto, TopicTable summarizes the latest and earlier documents for studying their contents and how these contents change with time.

MemeTracker [57] studies phrases and their frequencies of occurrence in news channels like blogs or information media portals. For visualizing these frequencies, MemeTracker uses a ThemeRiver approach [60], which applies a river metaphor to visualize changes in document contents over time. An example visualization for ThemeRiver is shown in Figure 4.2. ThemeRiver [60] visualizes the document frequencies[1] for predefined phrases over time. The document frequency of a specific phrase in each interval is mapped to the width of a flow, which "runs through time" from the left to the right side of the canvas. Flows of predefined phrases are plotted in different colors to make perceptual discrimination possible and are combined on top of each other to a river that flows through time. Space of the canvas is wasted (spent for uninformative background) whenever the total width of the ThemeRiver is small and the integration of text into narrow flows is difficult. ThemeRiver could be adapted to visualize the changing strength of topics over time as it is done for Topics over Time [41]. But enriching ThemeRiver with additional pieces of information is difficult due to space constraints; for example, the ThemeRiver visualization used in [41] does not provide enough space for printing topic headlines onto the visualized flows.

Other sophisticated approaches for visualizing document collections are Topic Map [62] and Topic Model Browser[2]. In contrast to TopicTable which aims at presenting topics and their evolution over time, Topic Map and Topic Browser summarize static collections. For example, Topic Map helps the reader in identifying sub-groups of similar

[1] number of documents that contain the word

[2] `http://www.cs.princeton.edu/~blei/topicmodeling.html`, section on "Corpus browsers based on topic models", March 26, 2012

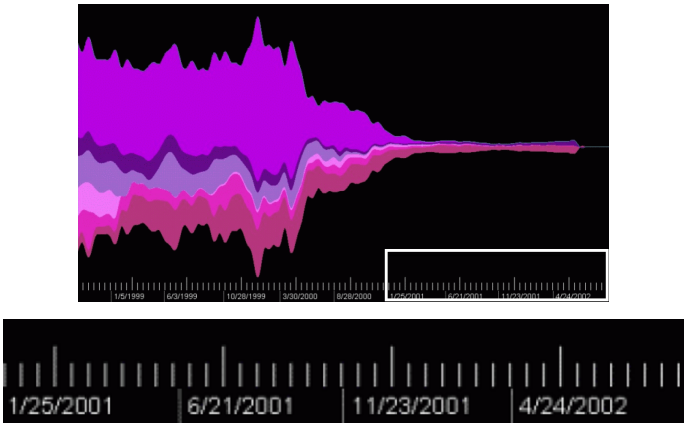


Figure 4.2: ThemeRiver on a few selected dot.com company stocks from January 1999 until April 2002 [61]. X-axis is timeline and framed part is printed enlarged at the bottom.

documents in large document collections. To this end, Topic Map uses topic modeling for dimension reduction and maps documents as represented by their high-dimensional word vectors into the 2-dimensional space for inspection. Topic Browser is an interactive visualization for overviewing learned topics by presenting their word distributions, topic strengths, and short summaries (headline and most likely words). It extends the simple box design by enabling the reader to interactively change among different websites. Each website presents a distinct pieces of information (topic strengths, word lists, topic distributions etc.).

4.2 Document prototypes

The concept of *document prototypes* is introduced as an interface between TopicTable and different approaches for learning patterns of words reflecting thematic subjects in streaming documents. TopicTable presents to the reader these document prototypes in combination with additional pieces of information which are useful for the interpretation of the presented document prototypes.

Document prototypes are condensed descriptions of the summarized documents. As such they abstract from specific documents. Consequently, the number of prototypes should be chosen to be substantially smaller than the number of documents. As a result, the reader has to inspect a few prototypes only instead of inspecting all documents. For summarizing the contents of documents, prototypes should give the reader hints for deducing thematic subjects which the documents are about. For example, a document prototype could be a list of words that often co-occur in documents of the summarized collection. To make a fast perception possible, these lists should consist of a few tens of words at most.

The concept of document prototypes makes it possible to separate the visualization technique from the learning method used to derive document prototypes. In general, arbitrary dimension reduction approaches might be suited for learning meaningful prototypes. Examples include principal component analysis, independent component analysis, non-negative matrix factorization, and topic modeling [62]. Probabilistic topic modeling

is used in this thesis for learning topics from a document stream. Topics capture patterns of words that often co-occur in different documents. As topics are distributions over all words of the vocabulary [15, 18] they are less suited for summarizing documents. Instead, document prototypes which consist of the N_{top} most likely words of each topic are derived from these distributions. As a consequence of deriving document prototypes from topics, the additional pieces of information which TopicTable visualizes are the strengths of and similarities among topics.

If one derives document prototypes, for example, from principle components determined by principle component analysis, the prevalence of a document prototype could be derived from the eigen value of the corresponding principle component. Similarities among document prototypes then could be determined as similarities among the underlying principle components.

4.2.1 Streams of documents

As explained in Section 3.2, a stream of documents is given by a sequence of document IDs $\vec{D}$ that are ordered by date of arrival. Arriving documents are pooled into batches $\vec{D}^1, \vec{D}^2 \ldots$. Topics are then learned for each batch from all documents that constitute this batch. These batches are defined by a window which slides over the streaming documents. At its i^{th} position, the sliding window covers some successive documents which constitute batch $\vec{D}^i$. An example is shown in Figure 3.1.

Parameters of the sliding window are (i) its size, and (ii) how far the window shifts to its next position. As described in more detail in Section 3.2, the size could be defined in units of either documents or time. For a given problem at hand, the definition that better fits the problem might be chosen.

4.2.2 Document prototypes over time

Document prototypes are derived from the topics of the topic models which have been adaptively learned; one model per batch. By inspecting the prototypes of successive batches, the reader gets an overview of the contents in each of these batches and how their thematic subjects change with time.

A subtle point of visualizing topics from successive batches is that these topics have to be meaningfully linked to each other for studying their evolution over time. This is a prerequisite of TopicTable because it presents prototypes that should be snapshots of similar thematic subjects over time along each row. In other words, the problem here is label switching. Three general approaches for coupling topics of successive topic models are described in the following. The topic model at point in time i is denoted by ζ^i and each model consists of K topics.

Post-hoc coupling means that models are learned for each batch independently of each other. Topics of subsequent models that are most similar to each other are matched afterwards. For example, among all topics of the model ζ^{i+1} the topic $1 \leq \bar{z} \leq K$ is most similar to the topic $1 \leq \hat{z} \leq K$ of model ζ^i. Then, topic $\bar{z}$ is identified as the follow-up topic of topic $\hat{z}$. This approach depends on the similarity measure between topics. An example of post-hoc coupling is the work of Mei and Zhai [63]. They learn independent PLSA models and use post-hoc coupling for the definition of threads of topics over time.

Coupling by initialization is taken by AdaptivePLSA [5] and the similar approach proposed by Chen and Chou [50]. Each topic model ζ^i is learned by a learning algorithm. Often, these learning algorithms are methods for numerical optimization and, hence, depend on the initialization of the model parameters. Coupling by initialization means that for learning the model ζ^{i+1} the start configuration of its model parameters are set equal to the learned parameters of the preceding model ζ^i. Consequently, the k^{th} topic in model ζ^{i+1} evolves from the k^{th} topic of the previous model ζ^i.

Coupling by priors is used by the Dynamic Topic Model [40]. Information about topics of successive models is transferred via priors for model parameters. A state space model makes the prior for topics of the later model ζ^{i+1} statistically dependent on the prior for topics of the preceding model ζ^i.

In this work, AdaptivePLSA [5], an extension of Probabilistic Latent Semantic Analysis to streams of documents, is used. AdaptivePLSA learns a PLSA model with K topics for each batch of a given sequence of batches. Afterwards, document prototypes are derived from the N_{top} most likely words per topic. The indices of the topics are associated to the derived document prototypes. Document prototypes with the same index over time are then presented along one row of TopicTable. The relation among these document prototypes along one row is meaningful as the underlying topics possess a meaningful connection. They evolve from each other over time as discussed in Section 3.6.5. Further details on AdaptivePLSA are discussed in Chapter 3.

4.3 Features of TopicTable

So far $\bar{N}$ batches $\vec{D}^1, \ldots, \vec{D}^{\bar{N}}$ have been constructed from a stream of documents. Applying AdaptivePLSA to these batches, one obtains $\bar{N}$ PLSA models $\zeta^1, \ldots, \zeta^{\bar{N}}$. Each topic model consists of K topics and a document prototype is derived from each topic.

TopicTable visualizes these comprehensible document prototypes in combination with additional pieces of information. Studying these document prototypes, the reader might reveal thematic subjects of the streaming documents and their thematic evolution.

TopicTable extends the tabular structure of the simple box scheme which is shown in Figure 4.1. For K document prototypes at $\bar{N}$ points in time TopicTable visualizes K rows and $\bar{N}$ columns. Rows correspond to document prototypes and columns correspond to snapshots of these prototypes over time. Hence, the cell (k, i) in row k and column i corresponds to the k^{th} prototype derived from PLSA model ζ^i. By arranging all k^{th} document prototypes of all batches along the k^{th} row, TopicTable visually establishes a correspondence among them. Aligning the k^{th} document prototypes over time is justified as AdaptivePLSA establishes a meaningful correspondence among the k^{th} topics over time. Arranging snapshots of related prototypes along the k^{th} row makes it easier for the reader to study these and to deduce their evolution over time.

TopicTable arranges four additional pieces of information. From background to foreground, these pieces are (i) local similarity between successive topics of each row, (ii) global similarity among all presented topics, (iii) relative strength of topics in each batch, and iv) emerging words. Figure 4.3 depicts how these pieces of information are visually presented by TopicTable. By displaying these pieces of information on top of each other from background to foreground, TopicTable uses the canvas efficiently. In addition, as TopicTable uses always one cell for each document prototype, TopicTable

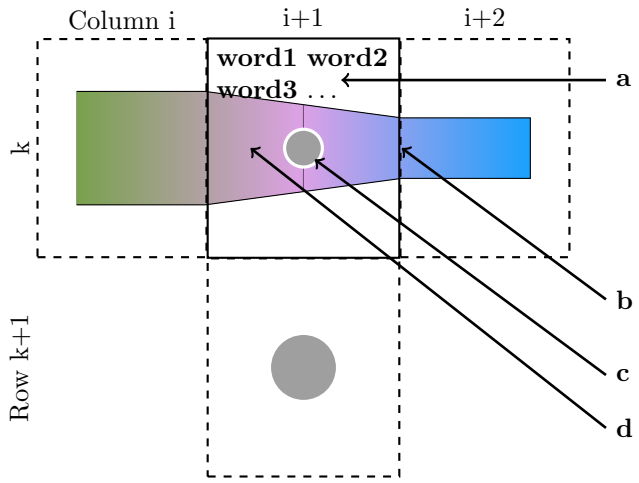


Figure 4.3: The cell in row k and column $(i+1)$ corresponds to the k^{th} topic which have been learned from the documents $\vec{D}^{i+1}$ of the stream $\vec{D}$. Features of TopicTable are: (a) document prototype (N_{top} most likely words of the underlying topic with new words being highlighted), and (b) the river in the background of each row has at each border between the $(i+1)^{\text{th}}$ and $(i+2)^{\text{th}}$ column a width that is proportional to the similarity between the k^{th} topic from batch $(i+1)$ and $(i+2)$, (c) the background circle whose radius is proportional to the relative strength (probability) of the corresponding topic, and (d) the color of the background river indicates similarities among all topics (similar topics have a similar color).

does not suppress but visually retain less dominant prototypes.

It should be noted that the similarities and relative strengths are determined for the underlying topics. This is a consequence of the decision to derive prototypes from topics of probabilistic topic models. If other approaches are used for deriving prototypes, these similarities and relative strengths will have to be determined in a different manner.

4.3.1 Local similarities between successive topics

First, TopicTable visually depicts how similar topics of one row between two successive batches are. Time periods in which successive topics of one row are relatively similar to each other might indicate life-times of thematic subjects. Moreover, if successive topics of one row are relatively dissimilar to each other, then they refer to relatively different patterns of co-occurring words. Visualizing local similarities between successive topics is a help for recognizing possible change of the thematic subjects these topics refer to.

For visualizing local similarities between topics, TopicTable uses the metaphor of a *river* that "flows through time". To this end, TopicTable associates each row with a river along that row. Narrow parts of the river are like watergates that strongly separate what comes before and what afterwards. These watergates indicate transitions during which the corresponding topic changes much; the similarity between the topic before and after this gate is low. In other words, the topic strongly evolves. As the width of the river in row k corresponds to similarities of successive topics with the same index, this river conveys the evolution of the corresponding k^{th} topic over time. Being drawn along the rows, background rivers visually connect cells of each row and thereby strengthen the

perception of the tabular structure of TopicTable.

TopicTable visualizes the rivers as colored straps along each row. The width of each river changes between successive cells to indicate watergates. Successive cells, say (k, i) and $(k, i+1)$, correspond to the k^{th} topic of model ζ^i and ζ^{i+1}, respectively. These topics are generalized Bernoulli distributions, which can be represented by two vectors of word-topic associations $\vec{x}_k^i$ and $\vec{x}_k^{i+1}$. The entries are probabilities of words of the respective vocabularies of batches $\vec{D}^i$ and $\vec{D}^{i+1}$. The more similar these two vectors are the more stable the corresponding topic is and the less it evolves.

For measuring similarity of two topics, TopicTable uses the cosine similarity between both vectors of word-topic associations. The cosine similarity takes values between 0 and 1; it is equal to 1 if both vectors point to the same direction (highest similarity) and it is equal to 0 if the vectors are orthogonal (lowest similarity). For computing the cosine similarity, both vectors $\vec{x}_k^i$ and $\vec{x}_k^{i+1}$ are embedded into the joint space defined by the union of the vocabularies of batches $\vec{D}^i$ and $\vec{D}^{i+1}$. Missing probabilities in both vectors are filled with zeros.

TopicTable maps the cosine similarity of successive topics $\vec{x}_k^i$ and $\vec{x}_k^{i+1}$ to the width of the background river at the border between cells (k, i) and $(k, i+1)$. The width of the river is proportional to the fifths power of the cosine similarity $\cos^5(x_k^i, x_k^{i+1})$; this emphasizes small differences among values close to 1. Further on, these transformed similarities are multiplied by $0.7 \cdot h$ with h being the height of the rows. This has the consequence that the width of the river is between 0.0 (similarity is 0) and 0.7 (similarity is 1) times the height of the cell. Keeping the maximal width of the background river below 70% of the row height enhances the perception of the rows.

4.3.2 Global similarities among all topics

Second, TopicTable visualizes the similarity between different topics at different points in time. This information eases perception of similar thematic subjects that are present in several separate periods of time.

Global similarities among topics are mapped to the color of the background rivers such that similar topics have a similar background color. TopicTable uses multi-dimensional scaling (MDS) for projecting all $\bar{N} \times K$ vectors of word-topic associations $\vec{x}_k^i$, with $1 \leq k \leq K$ and $1 \leq i \leq \bar{N}$, into a two-dimensional plane of the three-dimensional RGB color space. This two-dimensional plane is defined by all triples $(\mathrm{R}, \mathrm{G}, \mathrm{B})$ with R$\in [0; 255]$, G$=161$, B$\in [0; 255]$. A plot of this plane is shown in Figure 4.5.

MDS is computed with respect to the cosine distances[3] between the topics. As done for computing the local similarities, vectors of word-topic associations of any two topics are embedded into the space defined by joining the corresponding vocabularies. Missing word-topic associations are filled with zeros. As a result of MDS, each topic is mapped to a point in the two-dimensional plane and distances between points reflect the distances of the word-topic associations as good as possible. Next, the resulting points are linearly transformed such that all points lie inside the two-dimensional RGB plane, which is defined as $[0; 255] \times [0; 255]$. The point coordinates (x, y) are then directly mapped to a RGB color R$= x$, G$= 161$, B$= y$ which defines the color for the corresponding topic. The background rivers are colored with these colors such that the determined color for

[3] cosine distance is equal to 1 minus cosine similarity

the k^{th} topic of batch $\vec{D}^i$ is reached in the middle of the corresponding cell (k, i).

4.3.3 Relative strength of topics

Third, TopicTable visualizes the relative strength of the topics of each batch. This kind of information might be helpful, e.g., if a reader wants to study those document prototypes that correspond to the strongest topics, or, if a reader wants to inspect at a glance temporal changes of the strength of a topic.

For visualization of the relative strength of topics, TopicTable uses the topic probabilities $p^i(\text{z}{=}k)$ of each topic k from model ζ^i. All topic probabilities extracted from the i^{th} model sum to $1 = \sum_{k=1}^{K} p^i(\text{z}{=}k)$. A large probability indicates a strongly prevalent topic of batch $\vec{D}^i$. TopicTable visualizes these probabilities by circles which are depicted in the center of each cell. For enhancing perception of differences in the topic probabilities these are mapped to the circle areas in a nonlinear manner by the approach suggested by Cleveland [64]. In detail, the radius of the k^{th} circle in the i^{th} column is proportional to ${}^{p^i(z=k)^{5/7}}\!/_{\sqrt{2\pi}}$.

The circles are depicted in the background of the cell on top of the background river. Studying all circles of a column top-down gives a fast impression about what topics are the most dominant ones in a certain batch. Likewise, as each batch corresponds to a certain period of time, the reader gets an impression which topics are the dominant ones during which period of time. Inspecting the circles along a row, the reader might deduce how the relative strength of the corresponding topic changes with time. In addition, as circles are positioned in the center of the cells, they enhance the perception of the tabular structure of TopicTable.

4.3.4 Document prototypes

Last, TopicTable presents the document prototypes. In this work, these prototypes consist of the few most likely words of each topic. TopicTable lists these words in the foreground of the corresponding cells. Common choices of the number of listed words do not exceed some tens so that reading the word lists is of little expense. An experienced reader might deduce the thematic subjects from the top-10 words, a less experienced reader might need some more. Consequently, when applying TopicTable one can decide how many words should constitute the document prototypes.

As an additional help for perceiving changes of document prototypes, TopicTable highlights newly emerging words. Newly emerging word of the k^{th} prototypes in column $i + 1$ are those that are not part of the previous k^{th} prototype of column i. Words are highlighted by printing them in boldface. Many highlighted words might indicate that the thematic subject the topic refers to changes strongly while few or none highlighted words might indicate that the corresponding topic is relatively stable.

4.4 Influence of parameters

Four parameters mainly influence TopicTable and AdaptivePLSA. First, the size of the sliding window determines the number of documents a particular PLSA model is trained with. Likewise, it determines the time spans for which topics should be learned. These

time spans are given by the time stamps of the first and last document of the batches. The size of the sliding window needs to be adapted to the particular stream of documents whose contents should be summarized. For streams whose contents change fast, sliding windows of smaller size might be appropriate. On the other hand, if the contents do not change much, sliding windows of larger size might lead to the desired resolution of the summary.

A second parameter is how far the sliding window shifts to its next position and whether sliding windows at subsequent positions should overlap each other. If they overlap, then the learned topics will be more stable between subsequent batches. Topics between subsequent batches change more when the sliding window is shifted so far that some documents in between will not be covered by any window.

A third parameter is the number of topics K. This number affects the detailedness of the summarization of the streaming documents (see Figure 2.5). A reasonable choice is to set the number of topics K considerable smaller than the number of documents of a batch if a coarse summary of the contents of documents is desired.

Last, the number N_{top} of most likely words that constitute a document prototype is important. Too few words make it more difficult for the reader to deduce the thematic subjects from the visualized document prototypes simply because valuable context information is neglected. On the other hand, too many words befuddle the reader; resulting prototypes might represent to many minor thematic subjects.

Other parameters not discussed here are those which are necessary for AdaptivePLSA such as hyper-parameters or learning iterations of the EM algorithm.

4.5 Case studies

In this section, two case studies are presented which show how TopicTable helps for studying evolving contents of streams of documents. First, the SIGIR example, which was initially presented in Section 3.6, is continued. TopicTable was applied to topics of the PLSA models learned with the conference proceedings of the SIGIR conference between 2000 and 2007. A second subject of this case study is to investigate how effective the combination of AdaptivePLSA and TopicTable is for summarizing the contents of streaming documents. In a second case study, TopicTable was applied to documents published in the conference proceedings of the NIPS[4] conference from 1987 to 1999.

4.5.1 TopicTable for SIGIR documents from 2000 to 2007

Data preparation and parameter setting

Titles and abstracts of the documents published at the SIGIR conference between 2000 and 2007 were used. Preprocessing of these data is described in Section 3.6.1. A sliding window of size one year which was shifted by one year was used for determining batches. Consequently, each of the eight batches of documents contained all documents published in one year.

More details on how AdaptivePLSA was applies to the SIGIR batches are given in Section 3.6. With the help of AdaptivePLSA a sequence of eight PLSA models with

[4]Neural Information Processing Systems

$K = 5$ topics was learned. Document prototypes consisted of the $N_{\text{top}} = 25$ most likely words per topic.

TopicTable

The TopicTable for the SIGIR data is shown in Figure 4.4. The same topics are visualized with the simple box scheme in Table 3.11.

The first topic over time mainly refers to the subject of classification as the words *classif* and *classifi* are present across the whole row. Other subjects seem to be feature selection, support vector machines and collaborative filtering. These subjects are closely related to each other. Support vector machines could be used for collaborative filtering and a sub-area of support vector machines is feature selection. Indications for feature selection are the words *featur*, and *extract* and *select* highlighted in 2001 and 2003, and in 2005, respectively. The word *featur* constantly appears from 2003 to 2007 on ranks 4, 3, 3, 4, 1 indicating a strong emphasis on feature selection. As a clue to support vector machines, the word *vector* appears in 2000, 2001, 2003, 2006, and in 2007. In combination with *space* in 2000 and 2001 *vector* might indicate the vector space model for modeling document collections. Later, after a break in 2002 when *vector* does not appear, both *vector* and *machin* newly appear in 2003. In 2004 we find *support* together with *machin* and in 2007 *svm* occurs. These word combinations might indicate that support vector machines that are used for document classification is a later thematic subject of the first topic thread. The thematic subject on collaborative filtering is indicated by the word *filter* that appears first in 2000. After a break, this word re-appears and is present from 2004 to 2007; in 2004 and 2005, it remarkably appears in combination with *collabor*. The word combination *collabor* and *filter* is highlighted in 2004 making it an eye-catcher.

The second topic over time seems to capture thematic subjects mainly about evaluation of information retrieval algorithms. Words like *evalu*, *perform*, *rank* or *retriev* are always among the top three words. Additional thematic subjects might be identified by inspecting highlighted words. For instance, *trec* is highlighted in 2002 and is present until 2007 except in 2004. Trec is an acronym for Text REtrieval Conference[5] that provides test data-sets to evaluate approaches for information retrieval. Thus, the presence of the word *trec* strengthens the hypothesis that evaluation is a main subject which is captured by the second topic thread. Next, *feedback* is present in 2001 and from 2004 to 2007; it is highlighted in 2001 and 2004. The combination of *feedback* with *relev* might indicate that a thematic subject of some documents of the SIGIR stream is about *relevance feedback*. Another meaningful word is *video*, which is printed in boldface in 2003 and which additionally appears in 2004. This word might refer to ranking algorithms for video data.

Document clustering in information retrieval seems to be one main subject of the third topic thread. The background color between 2001 and 2000 is greenish and turns to brownish in 2003. This might indicate a change of the third topic from mining in databases to semantic techniques. Highlighted words between 2001 and 2003, which indicate data mining in databases, are *database*, *engine*, *search*, and *ir*. Other words, which indicate a subject on data mining, are *queri*, *retriev*, *search*, *question*, and *answer*. Later, the words *document* and *cluster* often appear among the top words of the

[5] `http://trec.nist.gov/`, March 26, 2012

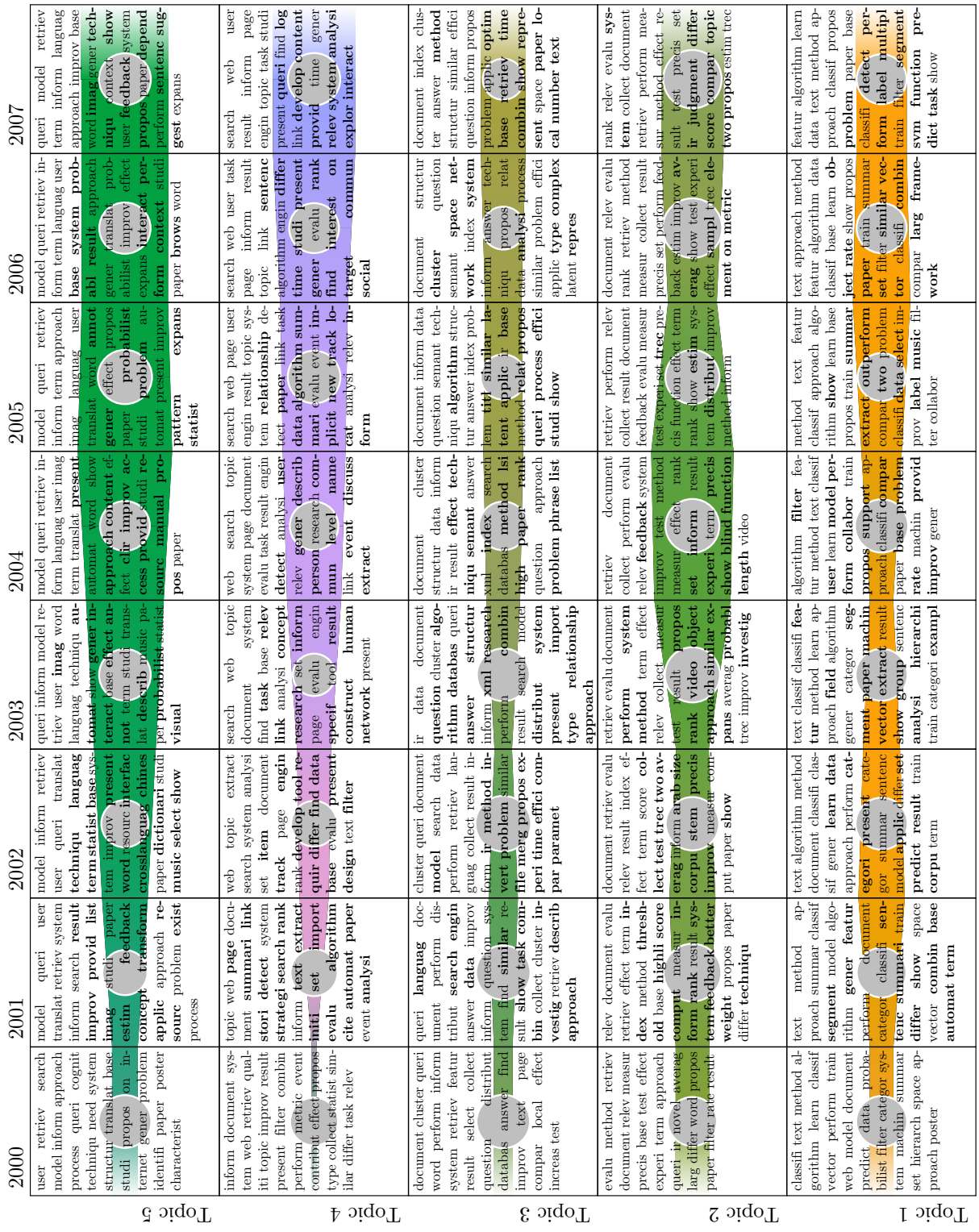


Figure 4.4: TopicTable for SIGIR data from 2000 to 2007. Document prototypes consist of the 25 most likely words of each topic. Numbers at the bottom indicate the number of documents per batch. It is recommended to use Adobe Reader for viewing Topic Tables because other PDF viewers might inadequately visualize shadings.

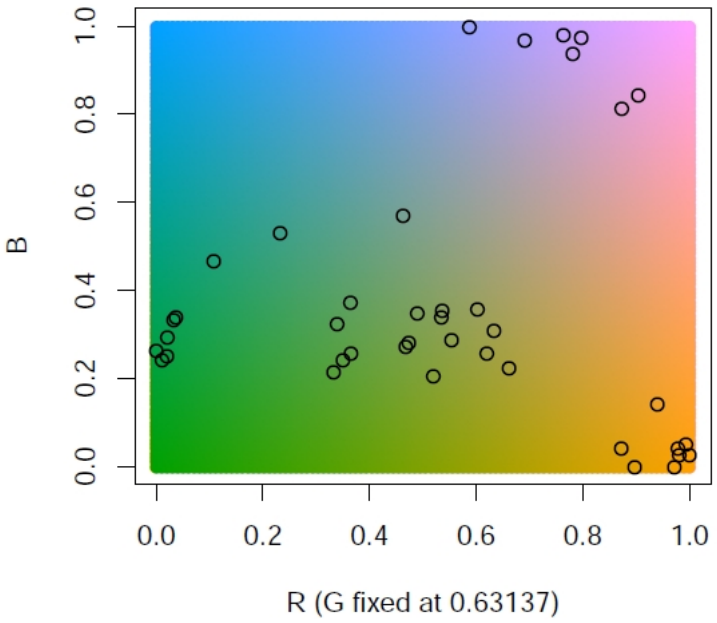


Figure 4.5: RGB plane which is used for color mapping of background rivers (the value of G is fixed at 161/255). High-dimensional vectors of word-topic associations of the SIGIR data were mapped onto this RGB plane via MDS. Each point belongs to one topic (cell) of TopicTable 4.4.

document prototypes. Highlighted words that provide clues to semantic techniques for document clustering are *distribut*, *lsi*, *techniqu*, and *latent*.

Web seems to be the main thematic subject of the fourth topic thread; *web* is almost always among the top-3 words. Boldfaced words that indicate the sub-areas (2001–2003) information extraction and link traversal are: *link*, *network*, *detect*, *extract*, *cite*, and *inform*. The following words, which appear from 2003 until 2005, might provide clues to a subject about user-specific web search: *user*, *person*, *name*, *web*, *search*, *relationship*. Last, in 2006 and 2007, boldfaced words that might indicate a sub-area of web search by exploiting social networks are: *interest*, *commun*, *social*, *interact*, and *user*. These thematic changes as captured by the fourth topic are emphasized by the changing color of the background river. The background river takes three different colors: pink (2001, 2002), lavender (2005), and light-blue (2006, 2007).

The fifth topic over time seems to refer to thematic subjects on multilingual IR and natural language processing. Although the word *translat* is present along the whole row, boldfaced words like *languag*, *crosslanguage*, *chines*, and *dictioranri* (2003) might indicate multilingual IR as a thematic subject until 2003. Later, natural language processing with a sub-area on automatic translation seems to be present (words *translat*, *word*, *automat*, and *languag* until 2005). Automatic annotation (*automat*, *manual*, *imag* between 2005 and 2007 and *anot* in 2003) and query expansion (*queri*, *retriev*, *expans* between 2005 and 2007) seem to be other sub-areas.

When overviewing the TopicTable and concentrating on the circles in the middle of each box, it seems that topics of the second thread are slightly dominating in each column, especially in the last one. Thus, it seems that evaluation of information-retrieval approaches is an important subject of documents published at the SIGIR conference. Obviously, in the metier of SIGIR, new algorithms are proposed and must be evaluated.

The other topics are often similarly strong. A reason for this equality could be that SIGIR is organized in tracks, each one having a different focus but their volumes are balanced. Another possible reason is that PLSA models, if not forced by imbalanced priors on topic-mixture proportions or by the data themselves, tend to learn topics whose relative probabilities are similar.

TopicTable visualizes colored rivers along each row. These rivers strengthen the perception of rows and thereby of the presented topic threads. The width of each river at transitions between successive cells indicates the similarity of successive topics with the same index. Some rivers, e.g., of the second or fifth thread, are relatively thick throughout the entire row. This indicates that these topic threads refer to a main stable thematic subject, e.g., evaluation of information retrieval algorithms (thread two), and multilingual IR and natural language processing (thread five). The overall smallest width of a background river is found at the transition of the fourth thread from 2000 to 2001. This indicates that the fourth topic considerably changes from 2000 to 2001. Consistently, the overall largest number (19) of new words per cell appears in the fourth prototype in 2001. In 2000, thematic subjects of the fourth document prototype seem to be diffuse; beside *web* words like *metric*, *statist* or *filter* are present. It seems that the fourth topic captures some diffuse subjects at the beginning and stabilizes later by focusing on web and its sub-areas.

The background river in row three indicates another relative strong change of the third topic between 2002 and 2003. Newly appearing words in 2003 like *databas*, *xml*, *realtionship*, *distribut* or *system* indicate this relative strong change. Hints such as a slim background river or a large number of newly emerging words are indications for a strongly changing topic. These indications are a help for determining thematic subjects and their evolution in a stream of documents.

Another helpful information is the similarity among all topics across time as indicated by the background color. The first, fourth and fifth topic have relative different background colors. This indicates that these topics refer to specific unique thematic subjects. The background colors of the second and the third topic seem to be closer to each other. This is a hint that these topics might share some particular thematic subjects. Indeed, the word *distribut* appears in the second prototype (in year 2005) and in the third prototype (in years 2000 and 2003). Remarkably, the color of both background rivers is greenish during these years. These findings might indicate that the second and third topic seem to share a subject on probabilistic modeling. As discussed earlier, the fifth topics seems to refer to probabilistic modeling, too. Again, the color of the background river of the fifth topic is greenish and words like *estim* (in year 2001), *statist* (in years 2002, 2003, 2005), and *probabilist* (in years 2003, 2005) are present.

Effectiveness

To show that reported prototypes are not artifacts but indeed summarize contents of streaming documents, 5 and 7 alien documents were added to the stream of SIGIR documents in 2003 and 2004, respectively. These alien documents, which have been randomly chosen from the journal *BMC Plant Biology*, account for about 7% of the documents of the changed SIGIR document stream in 2003 and 2004. As done with the SIGIR articles, the alien documents were reduced to their titles and abstracts, and they were subjected to preprocessing (stopword removal, Porter stemmer). Afterwards, eight

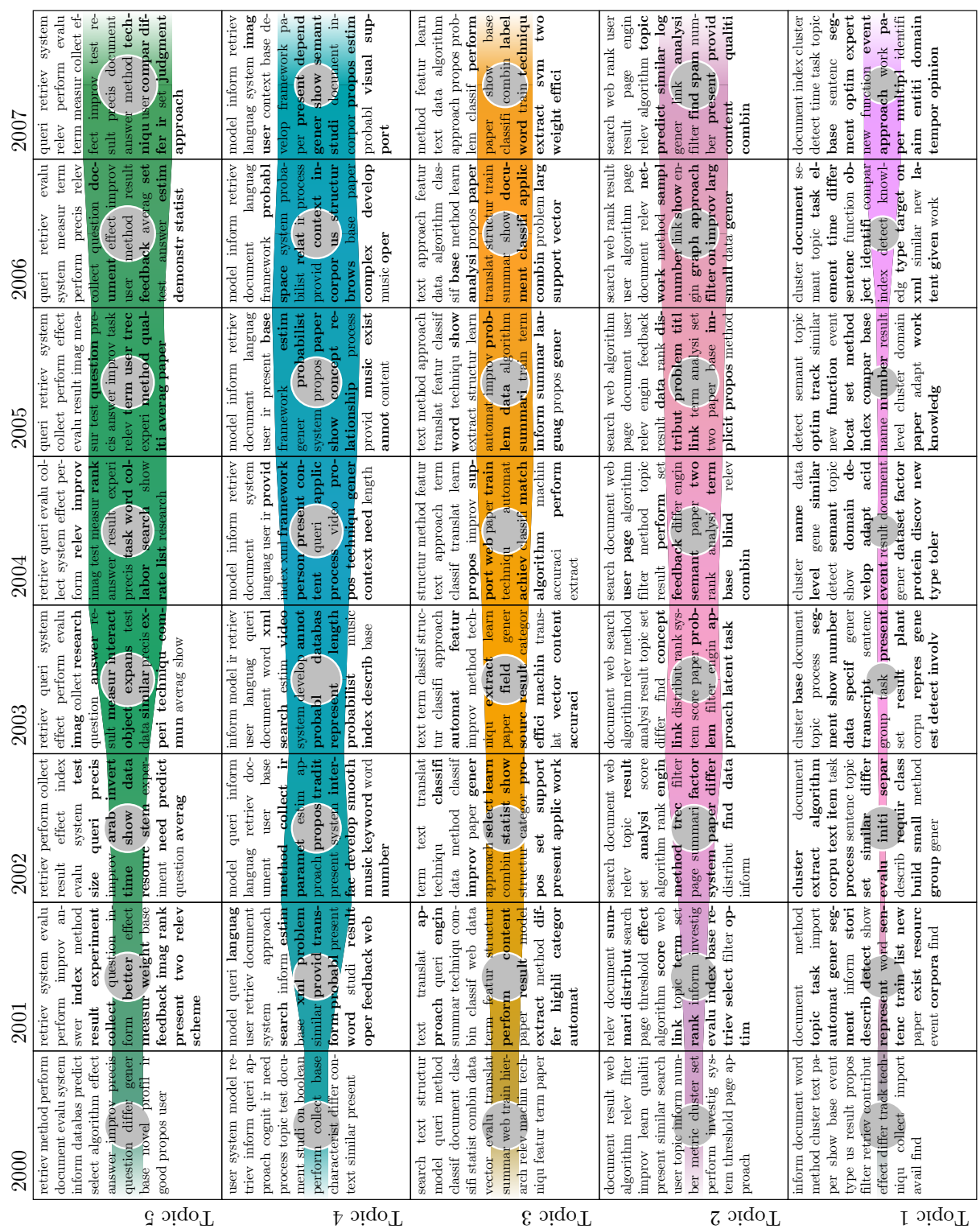


Figure 4.6: SIGIR data with alien documents, which have been added in 2003 and 2004, from the journal *BMC Plant Biology*. The first topic over time is the only one that is affected by added documents on plant biology (cf. Table 4.4).

PLSA models were learned from the extended stream with AdaptivePLSA as in case of the original stream of SIGIR documents.

The resulting TopicTable is shown in Figure 4.6. The first topic captures the alien documents. The background river of the first topic is clearly narrower than the other rivers. This indicates that the first topic changes much. The following words with biological context are present in 2003 and 2004: *base*, *transcript*, *plant*, *repres*, *gene*, *est*, *acid*, *factor*, and *protein*. These words give clues to the alien documents on plant biology. The presence of words from biological context within the listed document prototypes demonstrates the effectiveness of the combination of AdaptivePLSA and TopicTable for summarizing the thematic subjects and their evolution in document streams.

Last, the topics threads of both TopicTables shown in shown in Figure 4.4 (only SIGIR data) and in Figure 4.6 (with alien documents) should be compared with each other. The thematic subjects captured by the first, second, third, and fourth topic thread, which indicate main thematic subjects on clustering documents, web, classification, and language modeling, agree with the subjects as captured by the third, fourth, first, and fifth topic over time in Figure 4.4, respectively. Corresponding topics have different indices in both TopicTables because re-learning PLSA models might exchange indices due to random initialization. Transient thematic subjects might be captured by other or none topic thread after re-learning. The comparison of both TopicTables reveals that some minor thematic subjects appear together with the same main thematic subjects in both TopicTables. An example is the subject about support vector machines. This subject appears together with the main subject on classification in both TopicTables. Finding topics over time which capture the same main thematic subjects in both TopicTables indicates the robustness of the proposed approach for visually summarizing contents of document streams.

4.5.2 TopicTable for NIPS documents from 1987 to 1999

In a second case study, AdaptivePLSA was applied to documents from the NIPS conference proceedings from 1987 to 1999. The learned topics were visualized by TopicTable for visually summarizing these conference proceedings.

Data preparation and parameter setting

The NIPS data[6], which have been downloaded from Sam Roweis' website, were used for this case study. These data were preprocessed by removing stopwords and by applying the Porter stemmer. After the most frequent 50 words of the vocabulary were removed, the data did consist of 1740 documents, a vocabulary of 8621 words and about 1.7 million word occurrences in total. Sorted by year of publishing, these documents defined the stream of NIPS documents from 1987 to 1999.

A sliding window of size three years, which covered all documents published during three successive years, was applied for defining batches of documents. Successive batches overlap by one year. The time stamp of the last document per batch was used as a short annotation for each batch. For example, the batch which covered the time interval $[87, 88, 89]$ was denoted by 1989 ($\vec{D}^1$), $[89, 90, 91]$ (1991; $\vec{D}^2$), $[91, 92, 93]$ (1993; $\vec{D}^3$) and so on.

[6] `www.cs.nyu.edu/~roweis/data.html`, file `nips12raw_str602.mat`, April 5, 2011

As it was done in case of the SIGIR data, the number of topics K was set equal to 5 and document prototypes were defined to consist of the top $N_{\text{top}} = 15$ most likely words of each topic.

TopicTable

The resulting TopicTable, which summarizes of contents of the NIPS proceedings from 1987 to 1999, is presented in Figure 4.7.

At a first glace, the background rivers seem to be broader in relation to the cell heights when being compared to the background rivers of the SIGIR TopicTable (Figure 4.4). All background rivers in Figure 4.7 get relatively broad at least at some point in time whereas only the background rivers of the topic threads two, four and five in Figure 4.4 become that broad. The overlapping NIPS batches could be the reason; the SIGIR batches did not overlap. Overlapping batches might lead to more similar topics with the same index along each row as these have been learned from data sets that overlap partially. More similar topics along each row lead to larger widths of the background rivers. At a second glance, it turns out that each background river has its own particular color. Hence, the topics along different rows might refer to relative different thematic subjects.

After closer inspection of the first topic thread, one finds terms like *circuit* and *hopfield* in 1989 that might stand for specific architectures of artificial neural networks. Terms like *implement*, *chip* and *matrix*, which are present from 1989 to 1993, might indicate research dealing with development of specific hardware using concepts of neural networks. From 1993 to 1999, terms like *theorem*, *theory*, *minimum*, *complex*, *loss*, *converg*, and *bound* might refer to research on complexity and learning theorems of neural networks.

The background river of the second topic is narrower at the beginning and gets broader later on. This might indicate that the second topic changes more during transitions from 1989 to 1991 and from 1991 to 1993 compared to later transitions. In the beginning, terms like *cortex*, *simul*, *synaps*, and *oscillat* give clues to biological perspectives on neural networks. Later, in 1991 and 1993, the terms *visu*, *field*, *eye*, *object*, *locat*, and *region* appear. These terms might indicate research on image preprocessing using artificial neural networks. Later on, terms like *motion*, *movement*, *activity* occur which might represent research on processing of dynamic images, e.g., recorded by video cameras. Later, until 1999, TopicTable provides clues to a thematic subject on controlling robots (*head*, *orientat*, *motor*, *spaty* and *task*), and approaches of machine learning that make the robots respond (*human*, *respons*).

A general thematic subject of the third topic thread seems to be signal processing. At the beginning, terms like *synaps*, *fire*, *spike*, *signal* are present in 1989 and 1991. These terms seem to stand for biological research on how synapses process signals. Later, from 1991 to 1999, terms like *nois*, *filter*, *channel*, and *sourc* appear that might indicate research on implementing filters for signal processing using artificial neural networks. The continual change of the background color from pink/peach in 1989 to lavender in 1999 visually emphasizes the change of the third topic from a biological toward a technical perspective on signal processing.

From 1989 to 1999, at least one of the terms *class*, *classify*, or *recognit* is always among the top-2 words of document prototypes of the fourth topic thread. Hence, this topic thread seems to refer to pattern recognition with applications of artificial neural

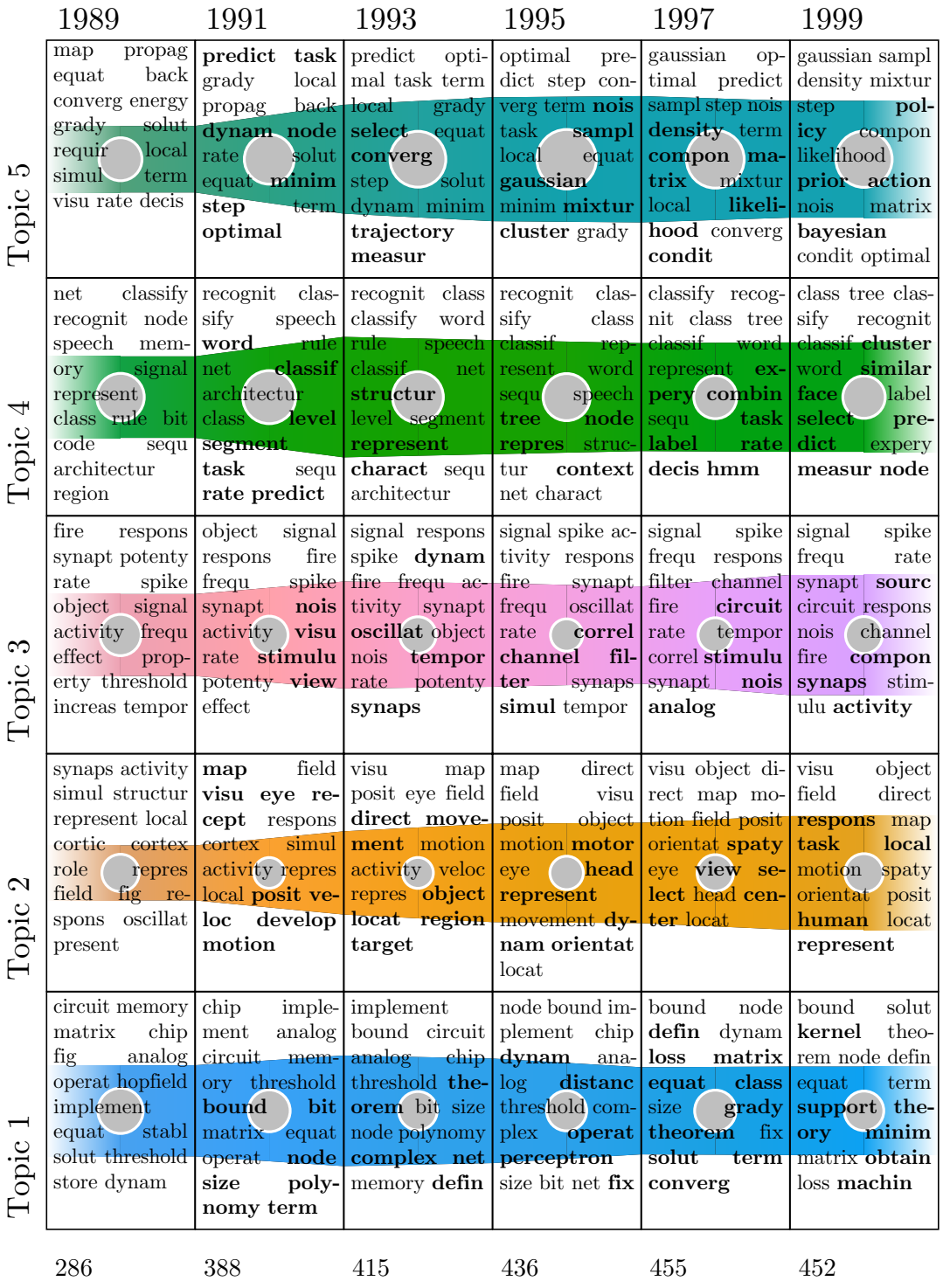


Figure 4.7: TopicTable for NIPS data from 1987 to 1999. Numbers at the bottom indicate the number of documents in each batch. It is recommended to use Adobe Reader for viewing Topic Tables because other PDF viewers might inadequately visualize shadings.

networks. Minor thematic subjects could be how decision rules are encoded by the structure of artificial neural networks (*rule*, *code*, *architecture* in 1989 and 1991), speech recognition (*speech*, *word* from 1989 to 1999), segmentation tasks (*segment* in 1991 and 1993), different designs of neural networks for classification (*tree*, *architecture*, *combin*, *hmm* present between 1995 and 1999), and face recognition (*face*, *cluster* in 1999).

The document prototypes of the fifths topic thread seem to mainly refer to a subject on learning approaches. Terms like *back*, *propag*, *optim*, *converg*, *energy*, *grady*, and *trajectory* are present between 1989 and 1993. These words seem to indicate a subject on learning neural networks with back propagation and related aspects thereof. Later, terms (*mixtur*, *cluster*, *likelihood*, *compon*, *sampl density*) which might refer to learning (Gaussian) mixture models occur between 1995 and 1999. In 1999, a subject on Bayesian learning seems to appear; the terms *bayesian*, and *prior* emerge.

Inspecting the gray circles across all cells, it becomes obvious that the fourth and fifth topic are dominating in 1989 and 1999. These topics refer to subjects on artificial neural networks, which are mainly used for solving classification problems, and learning approaches of artificial neural networks. Their dominance is not surprising as artificial neural networks have been a very active research field of the NIPS community in the early 1990's.

4.6 Conclusions and future directions

TopicTable is a visualization technique for studying the evolution of contents in a stream of documents. TopicTable presents document prototypes that are lists of a few words which reflect thematic subjects of the streaming documents. TopicTable can be used in combination with any method for the exploratory analysis of streams of documents. Examples are analytical methods like principle component analysis, discriminative methods, and probabilistic methods such as topic models which are used in this work.

Beside document prototypes, TopicTable visualizes four pieces of information which are helpful for deducing the thematic subjects from the document prototypes and their evolution. These pieces of information are: local similarities between two successive topics presented in the same row of TopicTable, global similarities among all topics of the entire TopicTable, strength of topics, and newly emerging words.

In combination with AdaptivePLSA, TopicTable was applied to the SIGIR conference proceedings from 2000 to 2007. Several thematic subjects like document clustering, evaluation, web and classification could be identified from the document prototypes. With respect to time, TopicTable indicates that, for example, support vector machines are a later subject appearing from 2004 to 2007. TopicTable gave valuable hints on a few alien documents about plant biology which have been merged into the SIGIR stream of documents. This demonstrates that TopicTable is indeed effective in providing a comprehensible summary of the contents of streaming documents.

In a second case study, TopicTable was applied to the NIPS conference proceedings from 1987 to 1999. TopicTable gave clues to different thematic subjects of the NIPS articles. Examples include image processing with artificial neural networks, biological neural network, mixture modeling, and Bayesian learning. Thematic subjects on learning artificial neural network and applications to classification, regression and prediction are the most dominant subjects. The background rivers of all topic threads are relatively

broad what indicates that the thematic subjects of the topic threads do not change substantially between successive years. A reason could be the small number of five topics. These might focus on different major NIPS subjects which changed only slightly between successive years. The focus of the learned topics on the different subjects is visually supported by the different colors of the background rivers. Hence, the additional pieces of information and their interplay might be helpful for deducing thematic subjects and their change with time from the presented document prototypes.

As online streams such as news channels or message services will become omnipresent in the future, tools for the exploratory analysis of these streams will become more important. TopicTable can be extended in several directions. For example, the diversity of contents of a stream of documents might change over time. As a result, the optimal number of topics needed for an comprehensible overview might change as well. Topic models and visualization techniques need to be extended such that they are capable of adapting the number of topics to a changing thematic diversity of streaming documents.

Another future direction is to cope with different complexities of thematic subjects. What is the optimal number of words for deducing a certain thematic subject from a topic? If methods are available which are capable of determining an optimal number of words to be presented, visualization techniques might be extended such that they represent different topics with document prototypes of different sizes. Beside the optimal number of words, it is also an open question which words should be presented to the reader in order to ease deducing thematic subjects from these words.

Third, the color mapping used by TopicTable can be optimized for human perception. Multi-dimensional scaling maps high-dimensional vectors of word-topic associations into the two-dimensional RGB plane as shown in Figure 4.5. This mapping takes into account only relative distances of the coordinates of the projected points. For the perception of differences in color, absolute coordinates should be taken into account, too. Two points with the same distance might result in different perceptions of the difference between colors they refer to. This is true, for example, when the two points lie in the middle of a green area or at the border between a green and a blue area of the two-dimensional RGB plane. Thus, color mapping could be enhanced in order to better map differences between topics to perceptual differences between colors.

Next, interactive visualizations are especially useful for exploratory analyses as the reader might better interpret document prototypes by own interactions with the visualization tool. Similar to Topic Browser, TopicTable could be extended into an interactive visualization application. Interactivity would make the integration of further information about the learned topics possible. For example, an interactive TopicTable could highlight all occurrences of a word over which the user moves the mouse pointer. This would help to identify terms which several prototypes have in common or terms which are specific for single prototypes. Another useful interaction could be to let the reader reset the number of words displayed for particular document prototypes. A larger number of listed words could be helpful for the interpretation of particular prototypes.

Last, a picture is worth a thousand words. The deduction of thematic subjects from word lists could be supported by presenting pictures that agree with the thematic subjects to which the word lists refer. To this end, topic models are needed which are capable of determining pictures that well agree with the thematic subjects of their topics.

Chapter 5

Exploring the semantic miscellany of social tags

Social tags are descriptive keywords which are assigned to online resources of the Social Web such as pieces of music of the online music platform `last.fm` or photographs of the online platform `flickr.com`. The users utilize social tags for managing and organizing online resources and for expressing opinions about these resources. As social tags are defined by the users themselves and as the same tag might be used by different users, meanings that the users associate with a particular tag could be a semantic miscellany. This chapter deals with exploring and clarifying the possibly many meanings of a social tag with the help of topic models.

Recently emerged Social Web services enable users to upload, manage, and share own resources. This chapter particularly deals with the two systems Bibsonomy[1] and CiteULike[2]. Both systems are used for managing and sharing bibliographic data sheets which mainly refer to scientific writings such as journal and conference papers. Such bibliographic data sheets, which are called bookmarks, contain different pieces of information about the documents they refer to, e.g., titles, authors, and abstracts. Bookmarks of these systems are mainly contributed by the users themselves as depicted in Figure 5.1 (left).

Bibsonomy and CiteULike make possible managing and organizing bookmarks with the help of social tags. Users of Bibsonomy and CiteULike might assign social tags to *own and others'* bookmarks as visualized in Figure 5.1 (right). For example, tags assigned to the bookmark referring to the paper "Latent Dirichlet Allocation" [15] (bookmark 4 in Figure 5.1, right) could be *topicmodeling* as this paper is about topic modeling, and *inference* as a user might firstly think about parameter inference, and yet another user might firstly think of Bayesian modeling and assigns the tag *Bayesian* to this bookmark. As Bibsonomy and CiteULike manage, beside bookmarks, social tags, these systems are called collaborative tagging systems.

In recent years, the phenomenon of social tagging started being investigated intensively. In collaborative tagging systems, tags are defined by the users themselves and all users might use all tags. This gives users the most freedom for tagging and, hence, for organizing and managing bookmarks, what is desirable as managing and organizing are the main purposes of social tags. Quoting Golder and Huberman ([65], p. 200, 203):

[1] `bibsonomy.org`, March 29, 2012
[2] `citeulike.org`, March 29, 2012

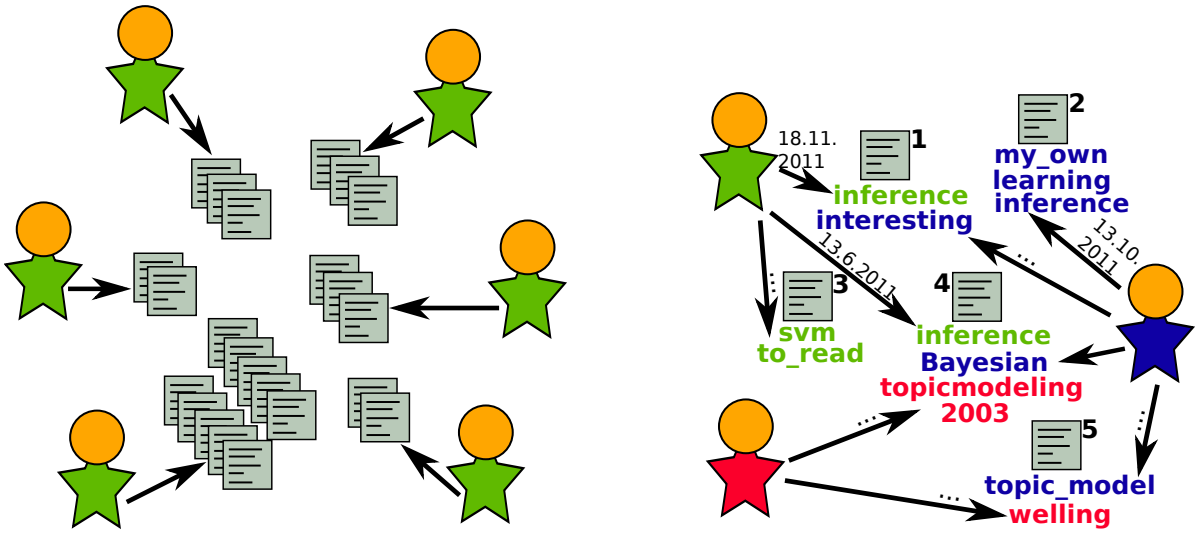


Figure 5.1: Collaborative tagging system: **left**) Users supply resources like bookmarks. All resources are publicly visible. **right**) Users assign tags to own resources and resources of other users. Assigning a tag to a resource is called a tagging event which is described by a tuple of the tag, the user ID, the resource ID, and the date.

"tagging is fundamentally about sensemaking", it is "an act of organizing through labeling, a way of making sense of many discrete, varied items according to their meaning."

Giving users the most freedom for defining and using own tags might result in unclear and semantically ambiguous tags. For efficient use, an ideal tag has a unique meaning that users, who take the tag at face value, are able to infer easily. The proliferation of tags that are semantically overloaded has severe implications: tag recommendation becomes more challenging, and the relationship between a tag and a bookmark already annotated with it might become questionable. As such, semantically ambiguous tags counteract the main purpose of tagging: making sense.

Sources of semantic ambiguity of tags are manifold. Huber and Goldman ([65], p. 199) state that

> "tagging systems [...] are beset by many problems that exist as a result of the necessarily imperfect, yet natural and evolving process of creating semantic relations between words and their referents. Three of these problems are polysemy, synonymy, and basic level variation."

Another source of ambiguity is that a particular tag might be used by many users who interpret this tag in different ways. Hence, with such a tag they annotate bookmarks that might refer to thematically different documents. As a result, the tag has multiple semantics [66].

Furthermore, as tagging is a permanent social activity of different users, semantic meanings of tags might change with time. Changing meanings could be counterproductive to an efficient usage of tags. The ultimate implication is that a user may fail to find the bookmarks associated with a specific meaning, including the own bookmarks, because the tag originally fitting the one meaning well now has multiple meanings, some of which different to the original meaning could have become the most prominent ones. Taking as a fact of matter that it is rather unsatisfactory for users to fail in finding the

bookmarks they look for, the following three situations might be disadvantageous for users of collaborative tagging systems.

1. Tags whose face value makes it hard for a user to infer their semantics. What could the Bibsonomy tag *v1002*[3] be about?

2. Tags with multiple semantics, some of which a particular user is not aware of. A user might want to tag own resources by most specific tags but is unaware of the (many) different meanings of the chosen tags. An example could be the Bibsonomy tag *network*, which is used for genetic as well as for social networks.

3. Tags whose semantic substantially evolves with time. A user's bookmark previously annotated with such a tag might not fit well anymore to the new semantics of that tag.

On the other hand, semantic ambiguities and yet unknown meanings of social tags have a value in their own respect. Tags convey meanings about the resources they are assigned to. For example, by studying tags that have been assigned to a particular resource by other users, a user might gain new perceptions on this resource as well as new conceptions of these tags. Enabling the users to perceive different perceptions on resources and relations among resources helps the users to learn from the implicit collaborative knowledge. This knowledge is a great value of collaborative tagging systems: Golder and Huberman([65], p. 201) state "there is also opportunity to learn from one another through sharing and organizing information." In addition, clashes of interpretation of tags have the potential for gaining new insight about tags and bookmarks they are associated with. Again, Golder and Huberman ([65], p. 207) state that "information tagged by others is [...] useful to the extent that the users [...] make sense of the contents in the same way [...]." For making sense in the same way, a user has to discover and learn about the potentially many meanings of common tags.

All these situations have in common that particular users are not aware of all semantic meanings and conceptions that the community of users as a whole associates with a tag. Exploring the semantic miscellany of a social tag, these users might gain collaborative knowledge. For example, by (i) extending the own conception of the tag, by (ii) adapting the own habits of tagging with the tag to the newly discovered meanings, and by (iii) being able of better interpreting perceptions on resources which other users have expressed through their tagging activity. Being aware of the many meanings the community associates with a tag, these users might work with this tag more efficiently.

The approach, which is described in this chapter, is retrospective in the sense that it explores the resources to which a social tag in question has been assigned in order to explore the semantic miscellany of this tag. Tagging activities are temporal processes and meanings which users associate with tags might change with time, some of which remain short-lived, while others proliferate because many users find them useful. Hence, the thematic subjects and their evolution associated with a tag in question are modeled by applying AdaptivePLSA (Chapter 3) to the documents under this tag. In detail, topic modeling of the documents under social tags is applied in order to explore the semantic miscellany of social tags in two ways. First, it is used for tag sensemaking by identifying unobvious meanings of social tags. Unobvious meanings are off-mainstream

[3] `http://www.bibsonomy.org/tag/v1002`, April 19, 2011

thematic subjects under a tag which might originate if a user annotates with this tag documents, which are, with respect to their content, different from the majority of documents annotated with this tag. Second, it is used for visual tag sensemaking, i.e., summarizing and studying thematic subjects of documents a tag is assigned to.

The rest of this chapter is organized as follows. Related work is discussed in Section 5.1. Details about how contents under social tags are modeled are given in Section 5.2. Next, in Sections 5.3 and 5.4, the two approaches for detecting minor meanings under social tags and for visual tag sensemaking are introduced and case studies are presented. Last, the conclusions are given in Section 5.5.

5.1 Related work

Tagging has gained much interest in the last years since collaborative tagging systems have become popular. Two main research directions are (i) using social tags and information derived from social tagging to enhance Web search engines and recommender systems, and (ii) studying the structure and the semantics of social tags [67].

Miliceviz et al. have reviewed recommender systems for tag or resource recommendation in collaborative tagging systems [68]. An example of such research is the work of Schwarzkopf et al. who discuss how a person's *tag space*, namely the set of tags used for annotations, can be exploited to derive the person's profile and, in sequel, to personalize the services offered by a provider [69]. Marco de Gemmis et al. consider tags next to contents to infer user interests for enhancing the content-based recommender of a collaborative tagging system [70]. A variety of probabilistic models for modeling data from collaborative tagging systems is proposed by different research groups. Wu et al. use a mixture model approach to statistically derive emergent semantics of resources from social annotations [71]. By that, they enhance, for a given bookmark, recommendation of semantically related web bookmarks. For utilizing social annotations for recommender systems, Zhou et al. propose a topic model that models document generation and tag assignments to documents [72]. A very similar idea is taken by Si and Sun who propose an extension of the well known topic model Latent Dirichlet Allocation [15] for modeling social tagging with the goal of enhancing tag recommendation [73].

Most studies assume that tags are representative for users' interests and for thematic subjects of resources. Recently, this assumption started being questioned. Zanardi and Capra ([74], p. 51) state that

> "as tags are informally defined, continually changing, and ungoverned, social tagging has often been criticized for lowering, rather than increasing, the efficiency of searching, due to the number of synonyms, homonyms, polysemy, as well as the heterogeneity of users and the noise they introduce."

These authors propose *Social Ranking*, a method based on recommendation advances; it combines tags *and* contents to enhance the performance of a search engine. A more drastic observation comes from Vatturi et al. [75], who point out that the elimination of those tags that have been used frequently over a long time improves the performance of a tag recommender. A similar course of action is taken by Sigurbjornsson and Van Zwol; their recommendation algorithm assigns lower scores to those candidate tags that are very frequent, because "tags with very high frequency are likely to be too general for

individual photos" ([76], p. 331). Hence, it seems reasonable to recommend popular tags with caution. Nonetheless, shedding light on the semantics that people associate with such tags can lead to more reasonable decisions on how to treat them in a recommendation engine. More important, shedding light on semantics of popular tags is part of tag sensemaking by which users of collaborative tagging systems familiarize themselves with the different conceptions the community associates with the tags and with the meanings of tag assignments to resources.

The international Dagstuhl seminar 08391 on *Social Web Communities* started a working group on the subject of tag semantics across collaborative tagging systems. This group studied tags in `delicious.com` and `flickr.com` and proposed a measure of semantic similarity between the co-occurrence vectors of tags [77]. Heymann and Garcia-Molina build a semantic hierarchy of tags with the help of a hierarchical clustering algorithm that exploits the co-occurrences of tags on resources and the tag centrality in the tag similarity graph [78]. Eda et al. explore semantic relatedness among tags for constructing a taxonomy of social tags [79]. In [80, 81], the authors studied and compared tag similarity measures that have been proposed in the literature for the identification of synonyms and for the determination of hierarchical relations among tags. The authors thereby study different forms of so-called *semantic grounding*, i.e., the use of resources like WordNet[4], to assess the meaning of individual tags [80, 81]. Ireson and Ciravegna studied how ambiguous concepts like social tags can be assigned a formal meaning by considering additional pieces of information to which the ambiguous concepts are linked via social annotation [82].

Some approaches take the *names* of the tags at face value. This is obviously necessary for the consultation of WordNet to do semantic grounding. Assuming that the name of a tag stands for its meaning is often a reasonable approach. For example, Benz et al. found that *apple* is associated with the computer company in Delicious, while users of Flickr also associate the term with the fruit [77]. However, identifying the meaning of tags like *funny* or *toread* might be less straightforward. The work which is presented in this chapter less focuses on associations among tags. Its subject is to contribute to making sense out of single popular tags independently of other tags they co-occur with. This corresponds to the idea of *tag sensemaking*, as propagated by Golder and Huberman [65]. Helping users in making sense out of social tags is also the goal of Tesconi et al. who derive links among ambiguous tags and contents of the *Wikipedia* for the investigation of questionable tags [83]. Another approach for clarifying the meanings of ambiguous tags is suggested by Yeung et al.. They propose graph-based methods to disambiguate tags by exploring the tripartite structure of collaborative tagging systems (resources, users, tags) [84]. Similar to the aim of the methods proposed in this chapter, De Meo et al. aim at supporting users in their work with tagging systems. They propose a probabilistic model for measuring similarity among tags and for arranging groups of semantically related tags in a hierarchy [85]. Visualizing the hierarchical structure among tags helps users to find tags best expressing their needs and interests. A visualization technique for disambiguation of social tags is also proposed by Hassan-Montero and Herrero-Solana [86]. Instead of visualizing contents of resources an ambiguous tag refers to, as is done in this chapter by visual tag sensemaking, Hassan-Montero and Herrero-Solana extend tag clouds for inspecting relations among tags derived from co-occurrence

[4] A lexical database for English. `http://wordnet.princeton.edu/`

data of social tags.

For the discovery of the meaning(s) associated with a popular tag, the work which is presented in this chapter exploits the relationship among tags and contents associated with them. The study of contents to assess tag semantics is not new by itself, and is of course an intuitive step to follow, even when the resources are not of textual nature. Examples are the already mentioned works of [72, 73, 84]. Another example is the work of Moxley et al. who derive the semantics of tags assigned to Flickr pictures by identifying and analyzing the geographical coordinates of the locations shown in the pictures [87]. However, it should also be considered that the meaning(s) associated with a popular tag may change over time. Consequently, to help users in making sense out of particular tags, the here introduced approaches take into account how the contents, to which a tag has been assigned, change with time.

In this chapter, approaches are proposed for exploiting semantic miscellanies of popular tags. These approaches build upon the documents which are linked to these tags. However, contrary to the studies cited so far [76, 80, 81], the here proposed methods neither resort to tag co-occurrence nor to the face value of the tag names. The names of popular tags might not be particularly informative and considering only co-occurrences of different tags seems too restrictive for an exploratory analysis of the meaning of tags. The intention in this study is to assist a user with a retrospective analysis to explore the possible many meanings of particular tags. Thus, the proposed methods aim at analyzing and summarizing documents which are linked to this tag through the annotated bookmarks.

For monitoring tag semantics over time the here proposed methods build on AdaptivePLSA as presented in Chapter 3. Topic models such as PLSA [18] and Latent Dirichlet Allocation [15] are especially suited for exploratory analyses of document collections. They consist of topics which are learned in an unsupervised manner from the documents. These topics, which reflect patterns of words that often co-occur in documents, might often be interpreted by humans as thematic subjects of the summarized documents. AdaptivePLSA, which is an extension of PLSA to streaming documents, takes the terminology evolution in account. Capturing terminological evolution is indispensable for exploring the semantic miscellany of social tags because changes of tag semantics might be associated with the increasing importance of some words that were irrelevant or unknown in the past.

5.2 Contents under a social tag

For exploring the semantic miscellany of social tags, a retrospective approach is applied to the contents to which tags are linked. Resources of Bibsonomy and CiteULike are bookmarks which refer to documents. By tagging bookmarks, users express their opinions and perceptions about the documents to which these bookmarks refer. Hence, bookmarks establish the connection between tags and documents. As the contents of these documents shall be explored for tag sensemaking, in this work, the referenced documents are identified with the bookmarks. As a consequence, tags are understood as being assigned to documents instead of bookmarks. It should be kept in mind that Bibsonomy and CiteULike actually manage only bookmarks which refer to documents.

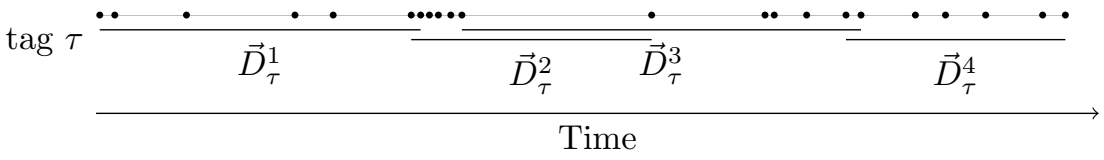


Figure 5.2: Stream of tagging events (small dots) corresponding to tag τ. By each tagging event a user has assigned tag τ to a document. The four positions of the sliding window of length $l = 7$ tagging events are represented by the horizontal lines. The sliding window shifts by $l_{new} = 5$ new tagging events (tagged documents).

5.2.1 Tagging events

For a given collaborative tagging system, the sets of all user IDs, document (content) IDs and tags are denoted by U, C, and Υ, respectively. Different document IDs might refer to the same document. A reason is that bookmarks that refer to the same document could have been added by several users. The collaborative tagging system then would assign different (document) IDs to these bookmarks (documents).

Collaborative tagging systems keep track of tagging events. In case of Bibsonomy or CiteULike, a tagging event is an action of annotating a bookmark – and hence the corresponding document – with a certain tag by a particular user at some point in time. Mathematically a tagging event is described by a tuple (τ, d, u, t) of a tag $\tau \in \Upsilon$, a unique document with ID $d \in C$, a user with ID $u \in U$, and a time stamp τ. Data $\mathbf{V}$, which describe tagging activities of a collaborative tagging system, are a set of N tagging events $\mathbf{V} = \{(\tau_i, d_i, u_i, t_i)\}_{1 \leq i \leq N}$.

5.2.2 Documents under a tag

The set of documents under a tag τ is $C_\tau = \{d_i \,|\, (\tau_i, d_i, u_i, t_i) \in \mathbf{V}, 1 \leq i \leq N, \tau_i = \tau\}$. The number of documents under tag τ, i.e., the cardinality of the set C_τ, is denoted by N_τ. By studying the documents C_τ, a user might gain knowledge about the thematic subjects of documents which are tagged with the tag τ. From these thematic subjects the user can deduce how other users interpret and use this tag.

5.2.3 Document stream under a tag

A collaborative tagging system is a place of intensive activity, where the community of users introduces new tags and associates new meanings with tags, while former meanings may become abandoned. New meanings under a tag emerge if users assign this tag to documents which are thematically different to former documents under the tag.

Detecting changes of meanings of social tags requires to take into account the dynamic nature of the tagging activities. For a certain tag, the time stamps of the tagging events define an ordering of the documents under this tag. Such an ordering of documents under the tag τ is represented by the sequence of document IDs of C_τ ordered by date. This sequence is denoted by $\vec{D}_\tau = (d_1, \ldots, d_{N_\tau})$, with $t_i \leq t_j$ for all $1 \leq i \leq j \leq N_\tau$. The sequence of document IDs defines a stream of documents under tag τ whose thematic subjects reflect the users' conceptions of this tag τ and their changes with time. An example of a stream under a tag is shown in Figure 5.2.

As described in more detail in Section 3.2, a sliding window might be applied to the stream of documents under a tag. This window covers l successive document IDs of the sequence of document IDs $\vec{D}_\tau$. The sliding window shifts by l_{new} documents at a time, i.e., the least recent l_{new} document IDs within the sliding window are forgotten when l_{new} new tagging events are recorded for tag τ. For the stream of documents $\vec{D}_\tau$ under tag τ the sliding window defines $\bar{N}_\tau$ batches of l document IDs $\vec{D}_\tau^1, \ldots, \vec{D}_\tau^{\bar{N}_\tau}$.

5.3 Minor semantics of social tags

One source of semantic miscellany of a social tag might originate from the tag usage of a particular user. The majority of users of a tag might agree upon their conceptions of the tag. But a particular user might associate a different, specific meaning with the tag. For example, most users tag documents about web or text search with the tag *search*. But a particular user assigns the tag *search* often to documents about AI search strategies, and so this user establishes a minor meaning under the tag *search*. This minor meaning might be unobvious for the majority of the other users. Being informed about the minor meaning under the tag *search*, the users may be interested in studying the documents that conform to this minor meaning. They may decide to use the tag *search* themselves for AI search strategies or to devise a new tag for this meaning.

Detection of minor meanings of social tags is a way of exploring their semantic miscellany. It should be stressed here that diversity in the way a tag is perceived is natural and often desirable. On the other hand, a user might well be interested in knowing whether other users consistently associate with a particular tag another meaning than the user oneself. This may or may not change the user's behavior but will certainly enhance the user's understanding of the tag.

5.3.1 Detection of minor meanings of tags in static document collections

Document IDs of documents annotated with the tag τ are denoted by C_τ and the subset thereof tagged by user u is denoted by $C_\tau(u)$. To analyze whether a user u', who tagged some documents $C_\tau(u')$ with tag τ, associates a minor meaning with tag τ, these documents $C_\tau(u')$ are compared to all documents C_τ under the tag τ.

A distance function, which is generically denoted by $\text{dist}(C_\tau(u), C_\tau)$, must be defined for this comparison. Later, in Section 5.3.3, two specific distance functions will be introduced. For the explanation of the algorithmic approach now, the generic formulation of the distance function is used.

The set of all user IDs of users who tagged with tag τ is U_τ which is defined as $U_\tau = \{u_i | (\tau_i, d_i, u_i, t_i) \in \mathbf{V}, 1 \leq i \leq N, \tau_i = \tau\}$. If a user $u' \in U_\tau$ associates a particular minor meaning with tag τ then it is hypothesized that the distance $\text{dist}(C_\tau(u'), C_\tau)$ is large in comparison to the distances $\text{dist}(C_\tau(u), C_\tau)$ corresponding to the other users $u \in U_\tau$ with $u \neq u'$. For quantifying how large this distance $\text{dist}(C_\tau(u'), C_\tau)$ is in relation to the other distances, the concept of the p-value is exploited. The p-value is the probability of observing an even larger distance than $\text{dist}(C_\tau(u'), C_\tau)$ for users who annotate with tag τ. The smaller the p-value for distance $\text{dist}(C_\tau(u'), C_\tau)$ is the less

likely it is to observe an even larger distance and the more likely becomes the conjecture that user u' indeed associates a particular minor meaning with tag τ.

The central question is with respect to which reference distribution should the p-value be determined. As the tag usage of a particular user should be compared to the average tag usage, the distribution of distances of all users U_τ is used as the reference distribution. Samples of this reference distribution are collected by the following bootstrap algorithm which is repeated b times ($1 \leq i \leq b$).

1. draw with equal probability a user $\hat{u}_i$ from U_τ
2. sample with replacement $|C_\tau(\hat{u}_i)|$ document IDs from $C_\tau(\hat{u}_i)$ and denote the sampled documents by $\hat{C}_\tau(\hat{u}_i)$
3. compute and store distance $\text{dist}(\hat{C}_\tau(\hat{u}_i), C_\tau)$

These sampled bootstrap distances are then used for determining the p-value for the distance $\text{dist}(C_\tau(u'), C_\tau)$ of a particular user $u' \in U_\tau$. This p-value, which is denoted by $p_{u',\tau}$, is equal to the fraction of bootstrap distances that are larger than the distance $\text{dist}(C_\tau(u'), C_\tau)$.

$$p_{u',\tau} := \frac{|\{\text{dist}(\hat{C}_\tau(\hat{u}_i), C_\tau) | \, \text{dist}(\hat{C}_\tau(\hat{u}_i), C_\tau) > \text{dist}(C_\tau(u'), C_\tau), 1 \leq i \leq b\}|}{b} \quad (5.1)$$

In summary, these p-values are determined for all users $u \in U_\tau$ in order to detect minor meanings of a tag τ. Then, the documents $C_\tau(u')$ are reported for each user $u' \in U_\tau$ for which the corresponding p-value $p_{u',\tau}$ is smaller then 0.05. The thematic subjects of these documents might represent minor meanings users associate with the tag τ. Users might inspect the contents of these documents for exploring potential minor meanings of the tag τ.

5.3.2 Two document representations

Different document representations might be more or less suited for detecting minor meanings under a tag. Consequently, two different representations are studied for their effectiveness.

The first representation is normalized word vectors. When the vocabulary is denoted by W and its size by $|W|$, then a normalized word vector $\vec{x}_d = (x_d^1, \ldots, x_d^{|W|})$ for document d is a vector of dimensionality $|W|$. Components x_d^j with $1 \leq j \leq |W|$ are relative frequencies of the j^{th} word of the vocabulary in document d. The components, which sum up to $1 = \sum_{j=1}^{|W|} x_d^j$, define a document-specific, generalized Bernoulli distribution over the vocabulary.

Usually, the size of the vocabulary might exceed several thousands of words. Hence, word vectors are high-dimensional representations of documents. As not all words occur in each document, some relative frequencies of words in a document are zero. Zero word probabilities might become problematic, e.g., if two generalized Bernoulli distributions should be compared to each other by the KL divergence. Hence, in order to prevent zero probabilities, a small pseudo count ϵ_W is added to each component of the word vectors. Afterwards, the word vectors are re-normalized.

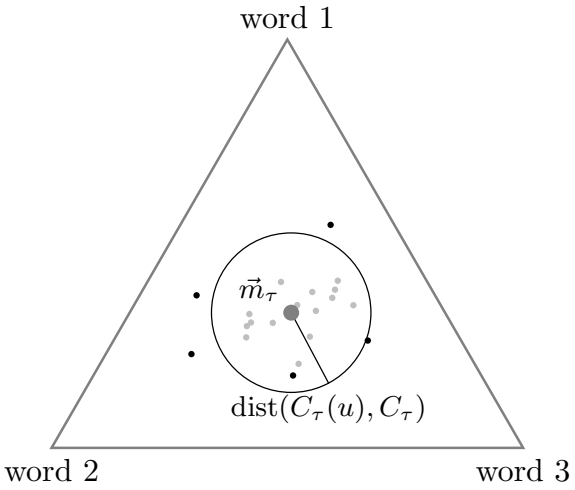


Figure 5.3: Scheme of distance functions which are used in this work. This kind of distance functions measure the mean distance (circle) of documents in $C_\tau(u)$ (black dots) to the mean representation $\vec{m}_\tau$ of documents C_τ (all dots). Here, documents are exemplarily represented as word vectors over a ternary vocabulary.

High-dimensional word vectors are sparse; a lot of their components are nearly zero. This sparseness might improve or worsen detection of minor meanings of tags. Hence, a second low-dimensional, less sparse document representation is used. To this end, a PLSA model ζ_τ (see Section 2.5 for details) is learned with the data of documents C_τ. Then, the topic-mixture proportions of the documents are extracted from the learned PLSA model. These low-dimensional vectors are used as a second document representation. The entries of each of these vectors define a document-specific, generalized Bernoulli distribution over topic probabilities of the corresponding document.

5.3.3 Two distances

The chosen distance function might affect the detection of minor semantics under social tags as well. Consequently, two distance functions are investigated for their effectiveness. Both determine the mean distances of documents in $C_\tau(u)$ to the mean representation of the reference documents C_τ. A scheme of the distance function is visualized in Figure 5.3. The mean representation of a document set C_τ is the mean vector $\vec{m}_\tau$ of documents C_τ. The j^{th} value m_τ^j of $\vec{m}_\tau$ is defined as $m_\tau^j = {}^1\!/\!_{|C_\tau|} \sum_{d \in C_\tau} x_d^j$.

The first distance function utilizes the cosine distance. The cosine distance is defined as follows

$$\text{cosDist}(\vec{x}_1, \vec{x}_2) \quad = \quad 1 - \cos(\sphericalangle(\vec{x}_1, \vec{x}_2)) \quad . \tag{5.2}$$

The cosine distance is smallest (equal to 0) if the vectors point into the same direction. It is largest (equal to 1) if the vectors are orthogonal. The first distance function, which determines the average cosine distance of documents in $C_\tau(u)$ to the mean representation

$\vec{m}_\tau$ of C_τ, is defined as follows

$$\text{dist}_{\cos}(C_\tau(u), C_\tau) \quad = \quad \frac{1}{|C_\tau(u)|} \sum_{d \in C_\tau(u)} \text{cosDist}(\vec{x}_d, \vec{m}_\tau) \quad . \tag{5.3}$$

The second distance function utilizes the KL divergence. The KL divergence comes from the field of information theory and measures the difference between two probability distributions. Formally, the KL divergence is defined between two distributions R and Q which are defined over the same discrete event space Y as follows

$$\text{KL}(R||Q) \quad = \quad \sum_{y \in Y} R(y) \log \frac{R(y)}{Q(y)} \tag{5.4}$$

and $Q(y) > 0$ for any y for which $R(y) > 0$. An interpretation of the KL divergence from information theory is the following. The KL divergence is the expected extra message-length per event $y \in Y$ that must be transmitted if a code that is optimal for a given (wrong) distribution Q is used, compared to using a code based on the true distribution R.

The second distance function, which measures the average KL divergence of documents in $C_\tau(u)$ with respect to the mean representation $\vec{m}_t$ of C_τ is defined as

$$\text{dist}_{\text{KL}}(C_\tau(u), C_\tau) \quad = \quad \frac{1}{|C_\tau(u)|} \sum_{d \in C_\tau(u)} \text{KL}(m_\tau||\vec{x}_d) \quad . \tag{5.5}$$

5.3.4 Adaption to streams of documents

So far, the detection of minor meanings of social tags has been introduced for static data collections. The meanings that users associate with a social tag might change with time. A stream of documents under a tag τ was defined. The contents of the streaming documents reflect the changing meanings which users associated with a tag. Applied to such a stream of documents, a sliding window defines a sequence of batches of documents $\vec{D}_\tau^1, \ldots, \vec{D}_\tau^{\bar{N}_\tau}$.

In the stream scenario, the approach for the detection of minor meanings is applied to each batch $\vec{D}_\tau^i$, with $1 \leq i \leq \bar{N}_\tau$. The documents of a user, who has contributed some documents to this batch, are denoted by $\vec{D}_\tau^i(u)$. Then, for each set of documents $\vec{D}_\tau^i(u)$ it is tested whether these documents contribute a minor meaning under the tag τ in $\vec{D}_\tau^i$. The bootstrap procedure is started anew for each batch and the bootstrap users are sampled from all users who have contributed to the batch $\vec{D}_\tau^i$. Documents are represented either by word vectors computed with respect to the vocabulary of batch $\vec{D}_\tau^i$ or by vectors of their topic-mixture proportions.

To obtain these topic-mixture proportions, a PLSA model ζ_τ^i has to be learned for batch $\vec{D}_\tau^i$. This is done by applying AdaptivePLSA (Chapter 3) which learns a sequence of PLSA models from the sequence of document batches. From these PLSA models, the low-dimensional document representations are extracted for the detection of minor meanings.

5.3.5 Experiments

First, details about the used data sets are given. Next, an evaluation experiment, in which an artificial minor meaning under a tag is induced, is described and results are presented. Last, in a real-world case study, minor meanings of two tags of the CiteULike system are explored.

Data from Bibsonomy and CiteULike

The experiments rely on data from the two collaborative tagging systems Bibsonomy[5] and CiteULike[6]. Data from the Bibsonomy system are easily accessible and well documented. Resources in Bibsonomy are bookmarks in the Bibtex format. For this work the cleaned dump of the Bibsonomy data set[7] was used, which was part of the data mining contest of the ECML/PKDD[8] conference 2009. This dump contains bookmarks and tagging events from December 31, 2005, to December 31, 2008.

Most bookmarks contain the abstract of the referenced document. These abstracts are used instead of the complete articles since full articles are often not accessible due to copyrights. In order to increase the number of bookmarks with abstract, the ACM Digital Library[9] was searched for missing abstracts. A few abstracts, which are written in German or French, were removed by the following simple heuristic. Bookmarks whose abstracts contained one of the words *der, die, das, ein, einer, eine, diese, dieser, dieses* and *sociaux*, were omitted. Abstracts of the remaining bookmarks were subjected to standard preprocessing techniques, i.e., removal of English stopwords and the Porter stemmer. Afterwards, all bookmarks whose abstract was shorter than 10 characters were omitted. Last, all tagging events except those that refer to one of the remaining bookmarks were removed.

A second data set is data[10] from the CiteULike system. To enrich this data set, missing abstracts were searched[11] in Medline[12], ArXiv[13] and CiteSeerX[14]. To this end, bookmarks were matched with data from these resources by specific CiteSeerX, PubMed, and ArchivX document-keys which were present in some bookmarks without abstract. By that procedure, 240.000, 9.000 and 13.000 abstracts from PubMed, ArchivX and CiteSeerX, respectively, could be matched. All abstracts were subjected to the removal of English stopwords and to the Porter stemmer. Bookmarks with an abstract shorter than 10 characters were omitted and all tagging events that referred to bookmarks other then the remaining bookmarks were neglected. The remaining abstracts of both data sets are called documents in the rest of this chapter.

---

[5] `bibsonomy.org`, March 29, 2012

[6] `citeulike.org`, March 29, 2012

[7] `www.kde.cs.uni-kassel.de/ws/dc09/dataset`, downloaded September 25, 2009

[8] Eurpean Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases

[9] `www.acm.org`, search was done in October 2009

[10] `citeulike.org/faq/data.adp`, downloaded April 21, 2010

[11] Data retrieval from the resources CiteSeerX, PubMed, and ArchivX, as well as data preprocessing was done by Bc. Ricardo Usbeck [88]

[12] `nlm.nih.gov/bsd/pmresources.html`, baseline data set from 1970 to 2009, accessible upon conclusion of a license agreement

[13] `arxiv.org/help/oa/index`, downloaded March 29, 2010

[14] `csxstatic.ist.psu.edu/about/data`, downloaded April 3, 2010

The popularity of a tag is measured by the number of tagging events by which this tag was assigned to a bookmark in the reduced data set. Popular tags are likely used by many users and so are likely semantic miscellanies. Thus, this study concentrates on the 40 most popular tags of Bibsonomy and CiteULike, which are listed in Tables 5.1 and 5.2, respectively. The CiteULike tag *no-tag* was neglected in the experiments as this is a technical tag used by the CiteULike system itself for resources without a tag.

Artificially induced minor meaning under a tag

For assessing the effectiveness of the proposed approach an artificial minor meaning under a tag was induced. The basic idea is to add off-topic documents to the documents under a tag. The tag under which a minor meanings should be induced is called the *target tag*. The tag from which the documents are take to establish the minor meaning is called the *source tag*.

An additional tag *bible_test* was added to the data sets Bibsonomy and CiteULike. Documents under this tag are 50 short excerpts from the Old Testament. These documents were subject to stopword removal and the Porter stemmer. The documents under the tag *bible_test* are assumed to be thematically very different to almost all scientific documents of Bibsonomy and CiteULike. Hence, the *bible_test* tag was used as the source tag to induce a minor meaning under the target tags listed in Tables 5.1 and 5.2.

In more detail, a test set of 200 documents was constructed for each target tag by randomly sampling $s \in \{2, 5, 10, 20, 35, 50\}$ documents from the source tag and $200 - s$ documents from the target tag with equal probability. All s documents from the source tag are assumed to have been contributed by one artificial user. The other users of the test sets are those who belong to the chosen documents from the target tags. The approach for the detection of minor meanings under tags was applied to all test sets individually. This resulted in determined p-values for each user of each test set. Although the document characteristics under different target tags might differ, the p-values are comparable to each other.

The proposed approach for detecting minor meanings under tags was applied to each of these data sets. In more detail, each combination of a data set (Bibsonomy or CiteULike), a value of s (6 possibilities), a distance function ($\text{dist}_{\cos}$ or dist_{KL}), and a document representation (word vectors or topic-mixture proportions with a number of topics $K \in \{5, 10, 20, 50, 100, 200\}$) was investigated. The obtained p-values were pooled across all 40 target tags for each combination. Each set of pooled p-values was used for classifying the corresponding users into the two classes of users (i) who contribute a minor meaning under the target tag (positive class), and (ii) who do not contribute a minor meaning. The ground truth for the positive class were the artificial users (tag *bible_test*), which have been added to each target tag. All other users were assumed not to add a minor meaning (ground truth of negative class). For assessing classification performance the area under the derived ROC curve (AUC) is reported. This area measures how well the proposed method discovers the artificial users as the users who have added a minor meaning. The resulting areas under the ROC curves are presented in Figure 5.4.

By varying $s \in \{2, 5, 10, 20, 35, 50\}$, it is investigated how the effectiveness of the proposed method depends on the fraction of off-topic documents. In addition, the effects of the two document presentations as discussed in Section 5.3.2 are assessed. The number

Table 5.1: 40 most popular tags of the preprocessed Bibsonomy data set.

	Tag	**# Tagging Events**	**First Usage**	**Last Usage**
1	algorithms	3688	2006-01-25	2008-12-21
2	genetic	3620	2006-07-07	2008-12-15
3	programming	3595	2006-03-09	2008-12-16
4	statphys23	1030	2007-06-20	2008-01-22
5	learning	768	2006-02-14	2008-12-28
6	web	580	2005-12-20	2008-12-28
7	software	579	2006-03-09	2008-12-27
8	ontology	567	2005-12-20	2008-12-29
9	information	564	2006-01-24	2008-12-27
10	evolution	534	2006-03-09	2008-12-09
11	folksonomy	532	2006-01-19	2008-12-21
12	social	526	2006-02-13	2008-12-27
13	design	520	2006-01-16	2008-12-28
14	tagging	518	2006-03-09	2008-12-27
15	wismasys0809	515	2008-10-22	2008-11-12
16	bibteximport	511	2006-03-07	2008-08-21
17	semantic	503	2005-12-20	2008-12-29
18	networks	480	2006-02-23	2008-12-27
19	theory	464	2006-01-24	2008-12-27
20	analysis	464	2006-01-31	2008-12-27
21	model	437	2006-01-08	2008-12-27
22	mrefs	418	2007-03-28	2008-12-04
23	systems	379	2006-03-09	2008-12-27
24	knowledge	372	2006-01-31	2008-12-27
25	semanticweb	359	2006-04-05	2008-12-20
26	book	358	2006-03-09	2008-12-27
27	network	357	2006-01-24	2008-12-27
28	evolutionary	355	2006-09-29	2008-12-06
29	management	347	2006-03-09	2008-12-27
30	2007	344	2007-01-29	2008-12-29
31	nlp	321	2006-07-26	2008-12-19
32	data	310	2005-12-20	2008-12-27
33	community	310	2006-02-15	2008-12-27
34	juergen	305	2008-02-26	2008-03-11
35	requirements	297	2006-03-24	2008-12-11
36	apob	293	2006-07-07	2006-07-07
37	toread	290	2006-04-05	2008-12-29
38	search	283	2005-12-20	2008-12-21
39	of	280	2006-01-06	2008-12-27
40	computer	277	2006-03-09	2008-12-27

Table 5.2: 41 most popular tags of preprocessed CiteULike data set. The technical tag *no-tag* was neglected in the experiments.

	Tag	# Tagging Events	First Usage	Last Usage
1	no-tag	43403	2004-11-05	2010-03-03
2	review	8081	2004-11-04	2010-03-03
3	evolution	3971	2004-11-11	2010-03-03
4	cancer	3274	2004-11-11	2010-03-03
5	network	2135	2004-12-19	2010-03-03
6	microarray	1936	2004-12-06	2010-03-03
7	protein	1935	2004-12-08	2010-03-03
8	bioinformatics	1817	2004-11-14	2010-03-03
9	networks	1781	2004-11-04	2010-03-01
10	structure	1781	2004-12-06	2010-03-01
11	fmri	1769	2005-02-01	2010-03-03
12	human	1608	2005-01-06	2010-03-03
13	model	1542	2004-11-12	2010-03-03
14	cosmology	1427	2004-12-27	2010-02-26
15	statistics	1328	2004-11-24	2010-03-01
16	methods	1274	2004-11-13	2010-03-02
17	expression	1187	2005-03-24	2010-03-01
18	genetics	1179	2004-11-11	2010-03-03
19	mirna	1177	2005-03-10	2010-03-03
20	memory	1123	2005-02-08	2010-02-26
21	software	1121	2004-11-12	2010-03-01
22	physics	1100	2005-03-04	2010-03-02
23	rna	1084	2004-12-06	2010-02-28
24	genomics	1074	2004-11-13	2010-03-01
25	development	1073	2004-11-10	2010-03-02
26	genome	1011	2004-12-21	2010-03-02
27	brain	999	2004-11-11	2010-02-28
28	gene	992	2004-12-19	2010-03-01
29	vision	990	2005-01-31	2010-02-28
30	theory	977	2004-12-30	2010-03-02
31	attention	976	2005-02-02	2010-03-02
32	proteomics	939	2004-11-13	2010-03-01
33	database	923	2004-11-11	2010-02-27
34	learning	905	2004-11-16	2010-02-18
35	breast	873	2004-11-30	2010-02-27
36	yeast	868	2004-11-04	2010-03-03
37	aging	868	2005-01-19	2010-03-02
38	ckd	863	2005-10-11	2010-02-19
39	simulation	861	2005-02-08	2010-03-03
40	quantum	837	2005-02-11	2010-02-23
41	dopamine	834	2005-01-12	2010-02-26

of topics $K \in \{5, 10, 20, 50, 100, 200\}$ was varied (several PLSA models with different topics were learned) in order to investigate whether the number of topics affects the effectiveness of the low-dimensional document representations. Further on, the effect of the two distance functions proposed in Section 5.3.3 on the detection of minor meanings is investigated.

Further parameters are the number of bootstrap iterations, which was set equal to $b = 200$, and the pseudo count, which was set equal to $\epsilon_W =$ 1e-6. The hyper-parameters of the PLSA priors were set to $\alpha = 1$ and $\beta = 1$ (cf. Section 2.5), and the EM algorithm for parameter learning was run for 20 iterations.

Influence of number of off-topic documents

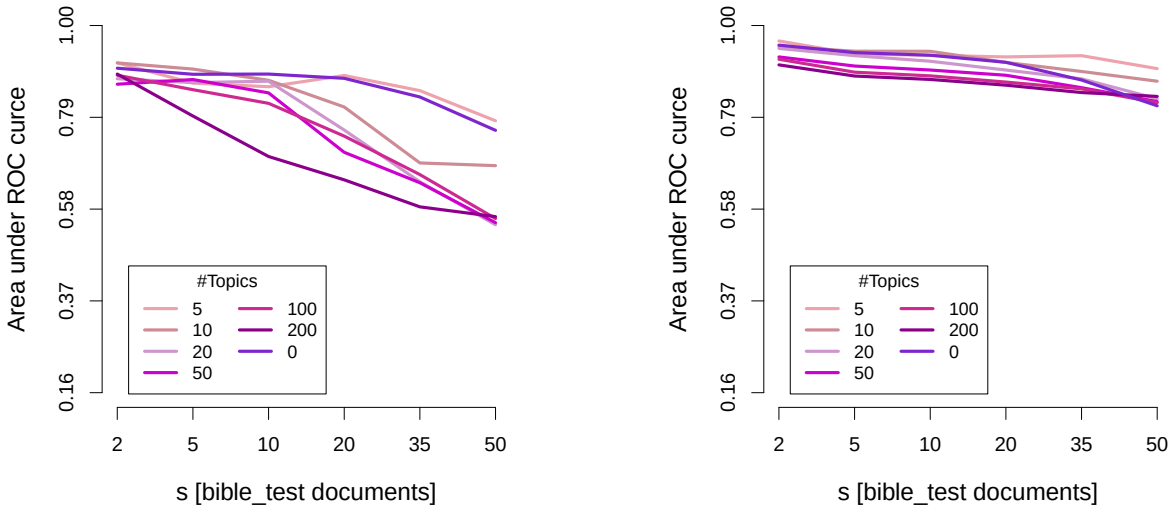
The obtained areas under the ROC curves (AUC) are presented in Figure 5.4. At the first glance it becomes obvious that all curves follow a similar characteristic: they decrease from left to right. That means, the reported AUCs, likewise the performance of the detection of the minor meaning, decrease with an increasing fraction of the *bible_test* documents. Two effects could be responsible for this observation.

First, while the fraction of *bible_test* documents increases from $^{2}/_{200}$ to $^{50}/_{200}$, the fraction of original documents under the target tags decreases from $^{200-2}/_{200}$ to $^{200-50}/_{200}$. This might result in a decreasing number of original users which are present in the constructed document test sets. Consequently, the artificial *bible_test* user will be selected more often during the bootstrap procedure and the *bible_test* documents will affect the bootstrapped reference distribution to a higher degree. This would lead to larger p-values for the *bible_test* documents which then are less likely detected as contributing a minor meaning under the target tag.

Second, all distances are computed in relation to the mean representations of the entire test sets. As the fraction of *bible_test* documents increases, their influence on this mean representation becomes stronger. Hence, the average distance of the *bible_test* documents to this mean representation might become smaller. Smaller average distances would result in larger p-values of the *bible_test* documents which lead to a worse detection of these documents as contributing a minor meaning under the target tag.

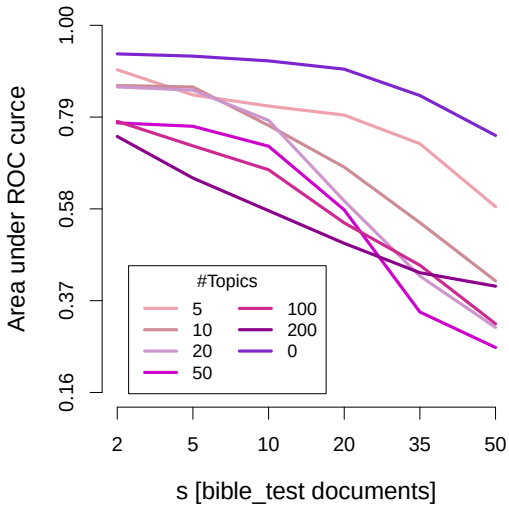
A worse recognition of the artificial *bible_test* user could also be desirable if this user has added a larger number of documents under the target tag. If this experiment had been run to its extremes by setting $s = 200$, then the only documents under the target tag would have been *bible_test* documents. These documents would have exclusively defined one main meaning of the target tag, namely a meaning associated to the Old Testament. In this case, it would have been wrong to detect this meaning as a minor one. The reason is that the main meanings of a social tag are defined by the majority of thematic subjects of documents associated to this tag. The higher the fraction of documents with a specific thematic subject, the more these documents define the meanings of this tag. Thus, it makes sense and it even might be desirable that the *bible_test* documents, if their fraction under the target tags becomes large, are not found to contribute a minor meaning anymore. These considerations rise the question, up to which relative size should a document set under a social tag be tested for contributing a minor meaning under a social tag.

The results which are presented in Figure 5.4 show that the proposed approach for the detection of minor meanings seems to work best when the fraction of documents,

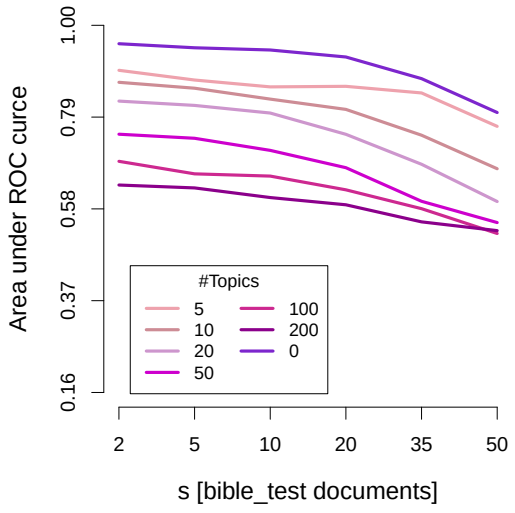


(a) Bibsonomy, cosine distance

(b) Bibsonomy, KL divergence



(c) CiteULike, cosine distance



(d) CiteULike, KL divergence

Figure 5.4: Area under ROC curves. Different plots correspond to different combinations of a data set (Bibsonomy on top, CiteULike at the bottom) and a distance function (cosine distance left, KL divergence right). #Topics = 0 encodes for representing documents by word vectors. The other curves correspond to representing documents by their topic-mixture proportions.

which contribute a minor meaning, is small. Thus, the proposed approach seems to be particularly well suited for recognizing sets of a few documents which contribute a minor meaning under a social tag.

Influence of document representation

The results presented in Figure 5.4 provide a clear picture about which document representation is better suited for the detection of minor meanings under social tags. Representing documents by word vector leads almost exclusively to larger AUCs, with two exceptions. The first exception is found for the combination of the Bibsonomy data, cosine distance, $s = 50$, and representing documents by topic-mixture proportions for $K = 5$ topics (Figure 5.4(a)). This combination gives almost as good results as using word vectors instead of the topic-mixture proportions. The second exception is found for the Bibsonomy data in combination with the KL divergence and $s \in \{35, 50\}$ (Figure 5.4(b)). Again, representing documents by vectors of their topic-mixture proportions for $K = 5$ and $K = 10$ topics gives similar AUCs as when documents are represented by their word vectors.

Word vectors could be better suited for the detection of minor meanings as they better represent occurrences of specific words. Stating this differently, representing documents by their topic-mixture proportions might increase the risk of smoothing differences among the documents with respect to their word compositions too much.

Interestingly, 5-dimensional topic-mixture proportions ($K = 5$) give similar AUCs compared to word-vector representations for the Bibsonomy data and $s \in \{35, 50\}$. An explanation could be that for $s = 35$ and $s = 50$ the *bible_test* documents are responsible for a fraction of $1/5.7$ and $1/4$ of all 200 test documents, respectively. These fractions are similar to $1/K$ with the number of topics $K = 5$. Hence, one could speculate that one of the five topics has exclusively specialized on the *bible_test* documents. The probability of this topic then should be relatively high for all the *bible_test* documents. At the same time, this probability should be relatively low for all other documents of the test sets. As a result, exploiting this topic as a feature which allows to well discriminate between the *bible_test* documents and the other documents could lead to the observed good classification performance.

To summarize these findings, word vectors seem to be better suited for the detection of minor meanings because the occurrences of specific words seem to be important. Merely by chance could topic-mixture proportions lead to a comparable well detection performance of minor meanings. The latter might happen when the fraction of documents, which indeed contribute a minor meaning, is similar to the fraction $1/K$ with K being the number of topics of the learned PLSA topic model.

Influence of distance function

Next, the effect of the two distance functions on the performance of discorvering minor meanings, is investigated. Average AUCs for the cosine distance (Figures 5.4(a) and 5.4(c)) tend to be smaller compared to the corresponding AUCs that are obtained when using the KL divergence (Figures 5.4(b) and 5.4(d)). These differences are more obvious for the values $s \in \{10, 20, 35, 50\}$. A reason could be that it is more challenging to detect minor meanings when the number of off-topic documents is larger as is the case for

$s \in \{35, 50\}$. The differences due to the utilized distance function are more pronounced under these more challenging experimental conditions. The choice of the distance function has only a minor effect on the detection accuracy of the induced minor meaning if the experimental settings are less challenging, i.e., if the induced minor meaning is more obvious as in the case of a small value of $s \in \{2, 5\}$. Over all, these findings indicate that the distance function which is based on the KL divergence seems to be better suited for the detection of minor meanings. Two reasons could be that the KL divergence seems to be more sensitive for detecting small differences between two probability distributions (document representations). In addition, the KL divergence was originally defined with respect to probability distributions and so naturally fits to the chosen document representations which actually are probability distributions.

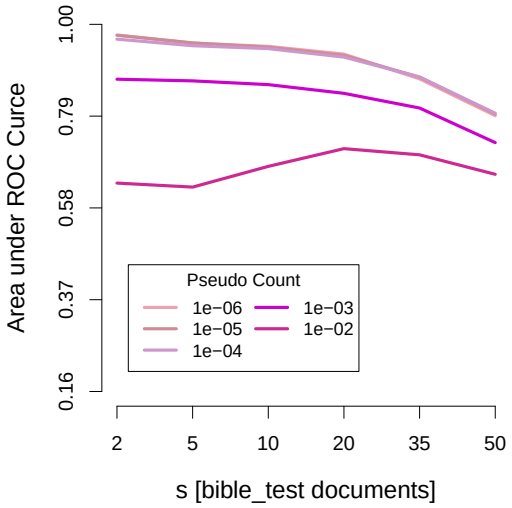
Influence of pseudo count

The findings indicate that representing documents by word vectors seems to give best accuracies of the detection of minor meanings under tags. So far, the pseudo count which was used for computing the word vectors was fixed at ϵ_W = 1e-6. The influence of this pseudo count on the effectiveness of detecting induced minor meanings is investigated with the help of a second experiment. This second experiment was similar to the first experiment but documents were represented exclusively by their word vectors and the pseudo count was varied $\epsilon_W \in \{$1e-6,1e-5,1e-4,1e-3,1e-2$\}$. The obtained areas under the ROC curves are presented in Figure 5.5.

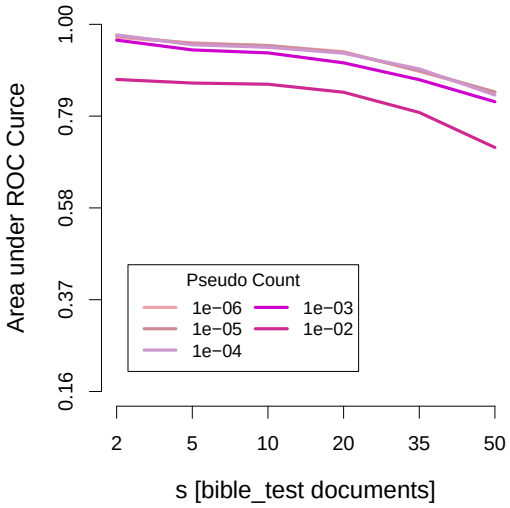
The obtained AUCs for pseudo counts in {1e-6,1e-5,1e-4} are often similar. The corresponding lines overlie each other in the plots shown in Figure 5.5. Further on, the AUCs for $\epsilon_W \in \{$1e-6,1e-5,1e-4$\}$ are larger than those for ϵ_W = 1e-4 with the exception of the combination of the Bibsonomy data, $s = 50$ and the cosine distance with one exception. This general pattern is more pronounced for the results which have been obtained with the Bibsonomy data. Increasing the pseudo count further to 1e-2 leads to even lower accuracies with the one mentioned exception. An explanation for this general observation could be that increasing the pseudo count leads to more strongly smoothed word vectors. As a result, differences among documents, which seem to be important for the detection of minor meanings, are worse represented by the word vectors. In summary, this experiments indicates that small pseudo counts $\epsilon_W \in \{$1e-6,1e-5,1e-4$\}$ should be preferred in order to keep differences among documents well detectable.

CiteULike case study

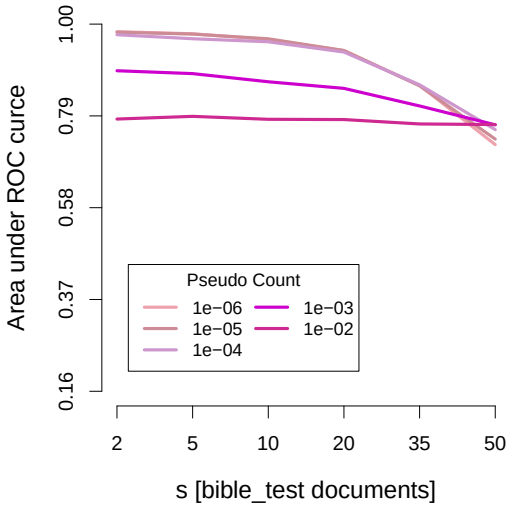
The contents under the two CiteuLike tags *structure* and *human* (cf. Table 5.2) were exemplarily analyzed. For detecting minor meanings over time, the proposed method was applied to the streams of documents under these two tags as described in Section 5.3.4. A sliding window was applied to these document streams such that each batch was of size $l = 600$ documents. The sliding window was shifted by $l_{new} = l \cdot 0.75 = 450$ documents such that successive batches overlap by 150 documents. The bootstrap procedure was repeated $b = 200$ times, the documents were represented by their word vectors, the pseudo count was set to ϵ_w = 1e-6, and the distance function dist_{KL} which is based on the KL divergence was applied. Document sets were reported as contributing a minor meaning if their determined p-value was lower or equal than 0.05.



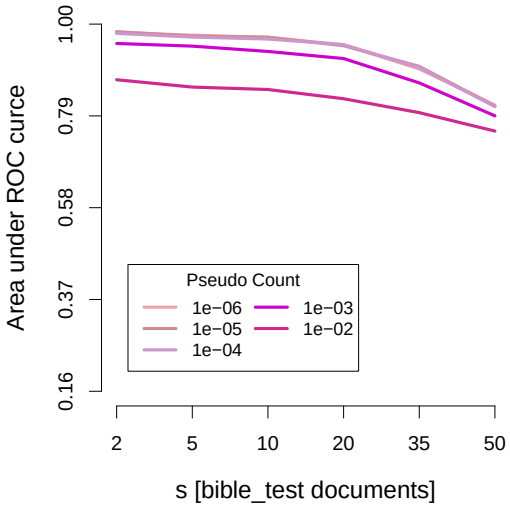
(a) Bibsonomy, cosine distance



(b) Bibsonomy, KL divergence



(c) CiteULike, cosine distance



(d) CiteULike, KL divergence

Figure 5.5: Influence of pseudo count ϵ_W on the detection of minor meanings under tags. In this experiment, documents were represented only by their word vectors.

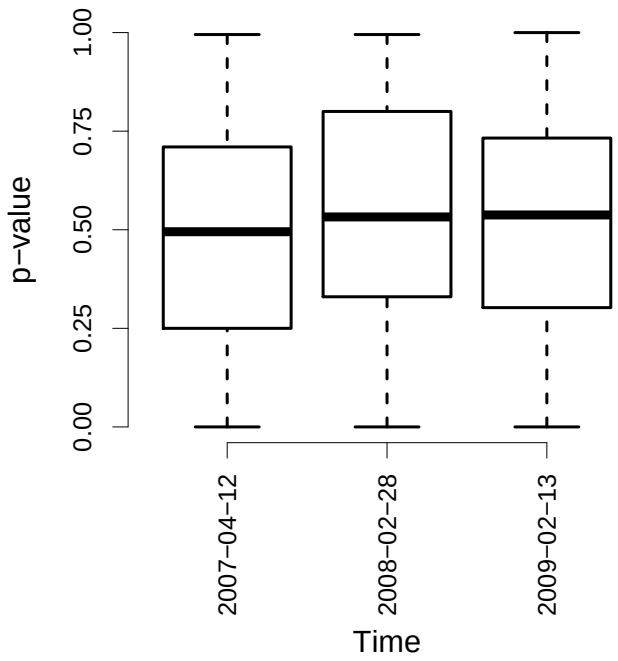


Figure 5.6: Overview of p-values of document sets under the tag *structure*. The sliding window defined three batches for the stream of documents under this tag.

Minor meanings under tag *structure*

The empiric distribution of the obtained p-values for all document sets of each of the three batches under the tag *structure* are visualized in Figure 5.6. A few p-values are at most 0.05. The corresponding document IDs are listed in Table 5.3. All but one of these reported document sets consist of one document only. This agrees with the tendency that smaller document sets are more likely to be detected as contributing a minor meaning. The titles of the documents of the set (1631098; 7520279) of batch 3 are "Suppressor mutations in Escherichia coli methionyl-tRNA formyltransferase: role of a 16-amino acid insertion module in initiator tRNA recognition" (doc ID 1631098) and "Crystal structures of wild-type p-hydroxybenzoate hydroxylase complexed with 4-aminobenzoate,2,4-dihydroxybenzoate, and 2-hydroxy-4-aminobenzoate and of the Tyr222Ala mutant complexed with 2-hydroxy-4-aminobenzoate. Evidence for a proton channel and a new binding mode of the flavin ring" (doc ID 7520279). This finding indicates that a minor meaning under the tag *structure* in the CiteULike data is molecular structures (crystal structures and formation of secondary structure of tRNAs) from the context of biochemistry.

Minor meanings under tag *human*

The constructed stream under tag *human* consists of three batches. The empiric distribution of the obtained p-values are visualized in Figure 5.7. Table 5.4 lists all reported document sets, which are supposed to contribute a minor meaning, in detail. One reported document set of the first batch is the one with document IDs 3548575 and 11785818. The titles of these two documents are "Auditory psychophysics: spectrotemporal representation of signals" and "Towards a measure of auditory-filter phase

Table 5.3: Summary of reported document sets from the stream of documents under the tag *structure*. Each row corresponds to one document set (all but one consist of one document only).

Batch ID	Date	User ID	P Value	Doc ID
1	2007-04-12	-	0.03	11133963
		-	0.04	astro-ph/0109350
		-	0	11690047
		-	0	10.1.1.12.4336
2	2008-02-28	-	0.035	10.1.1.12.4336
		-	0.03	10.1.1.14.2630
		-	0.05	10.1.1.11.2024
		-	0	10.1.1.14.4712
3	2009-02-13	-	0.03	17984963
		-	0.03	13898346
		-	0	10.1.1.14.4712
		-	0.015	17194605
		-	0.02	1631098; 7520279
		-	0	3680217
		-	0.05	0802.0485

response". These documents seem to induce a minor thematic subject about the human auditory system under the tag *human*. One reported document set of the second batch consists of the three documents 11123839, 15689531, 18046746. Their titles are "The primate cranial base: ontogeny, function, and integration", "Neandertal evolutionary genetics: mitochondrial DNA data from the iberian peninsula" and "Paranthropus boisei: fifty years of evidence and analysis". *Paranthropus boisei* was an early hominin which was first discovered by the anthropologist Mary Leakey on July 17, 1959, at Olduvai Gorge, Tanzania[15]. These documents seem to contribute a minor meaning about human ontogenesis and phylogenesis to the tag *human*. Another document set which consists of the three documents 10627087, 11153847, 12011790 of the third batch was reported as conveying a minor meaning. These documents seem to add a thematic subject about human odor from a social and psychological perspective. Their titles are "Rapid mood change and human odors", "Human olfactory communication of emotion" and "The scent of fear".

5.4 Visual tag sensemaking

Collaborative tagging systems are highly dynamic with respect to the way how their users interpret and use tags. Since users might change their interpretations of tags as well as new users join the system and newly interpret existing tags, meanings associated with social tags change over time. As the contents of documents a tag is associated to reflect the meanings of this tag, studying the evolution of these contents helps to unravel the semantic evolution of social tags. Motivations of a particular user to study the semantic miscellany of a tag are diverse. The tag might be totally unknown to the

[15] wikipedia.org, October 14, 2011

Table 5.4: Summary of reported document sets from the stream of documents under the tag *human*. Each row corresponds to one document set (some consist of one document only).

Batch ID	Date	User ID	P Value	Doc ID
1	2007-08-16	-	0	8710414
		-	0.03	16809528
		-	0.05	3548575; 11785818
		-	0.03	15821430
		-	0.045	17652657
2	2008-06-02	-	0.01	17395643; 14596797
		-	0.03	2934473
		-	0.015	10.1.1.12.7553
		-	0.025	9887022; 16452121
		-	0	17652657
		-	0.01	18096770
		-	0.045	17507175
		-	0.025	11123839; 15689531; 18046746
		-	0.025	17943116
		-	0.045	16352667; 16046292
		-	0.05	11153153
		-	0.025	10.1.1.10.4507
3	2009-07-15	-	0.015	11123839; 15689531; 18046746
		-	0.045	17943116
		-	0.025	10.1.1.10.4507
		-	0.04	16126811; 2214715; 11724665
		-	0.005	10.1.1.104.2324
		-	0.015	2300263; 1677640
		-	0.02	10627087; 11153847; 12011790

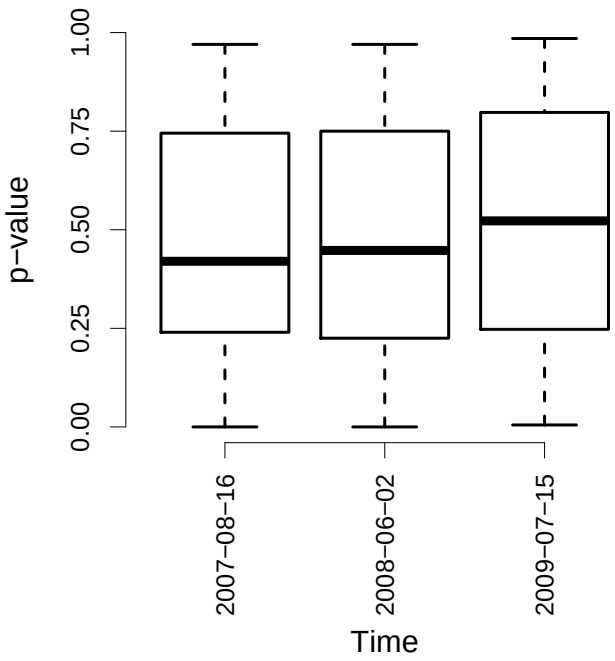


Figure 5.7: Overview of determined p-values for tag *human*.

user, i.e., the user is incapable of deducing the meaning of the tag from its face value. Another motivation could be that, although the user has an interpretation of the tag, the user might want to gain knowledge about further meanings which the community of users associates with this tag.

To facilitate studying the contents of documents under a tag, these contents and their changes over time need to be presented to the user in a comprehensible manner. By visually perceiving and studying the presented information about the contents under a tag, a user might deduce the diversity of meanings of a tag and their changes over time. This process of investigating the semantic miscellany of a tag by studying visualized pieces of information about the contents of documents under the tag is called *visual tag sensemaking*.

TopicTable (Chapter 4), which is used in combination with AdaptivePLSA (Chapter 3), is effective for summarizing and visualizing contents of document streams and their changes over time. Thus, this combination is especially suited for visual tag sensemaking. For a stream of documents, TopicTable generates a condensed and comprehensible overview of the document contents and their changes over time. Applying AdaptivePLSA and TopicTable to a stream of documents under a social tag provides helpful hints to a user for deducing the meanings users associate with this tag.

Beside studying thematic subjects associated with a tag, a user might also be interested in studying the own influence on a particular tag. Knowing which of the summarized and visualized contents are due to the own tagging activity, the user might concentrate on the other summarized contents for deducing yet unknown thematic subjects. To this end, TopicTable is extended in such a way that it visualizes the influence of a particular user on the presented topics.

5.4.1 User influence on a tag

The documents under a tag have been tagged by the entire community of users. TopicTable, which is used in combination with AdaptivePLSA, presents an overview of the contents of these documents under the tag. Through the own tagging activities, a particular user influences the contents under this tags and so influences the meanings of this tag. Thus, the fraction of documents under a tag contributed by a particular user is a measure of the user's influence on the meanings of this tag. Making perceivable this influence of a user on the semantic miscellany of a tag enhances the usefulness of TopicTable for visual tag sensemaking.

In the following, it is explained how a user's influence on the topics, which are presented by TopicTable, is measured. A sequence of document batches $\vec{D}_\tau^1, \ldots, \vec{D}_\tau^{\bar{N}_\tau}$ is defined by a sliding window which is applied to the stream of documents under a tag as described in Section 5.2.3. Afterwards, AdaptivePLSA (Chapter 3) is applied to these batches. AdaptivePLSA learns a PLSA model ζ_τ^i for each batch $\vec{D}_\tau^i$. It is made explicit that the models and their parameters are specific of a certain tag τ by adding the sub-script τ to the corresponding symbols. Each model has K topics which are indexed by the index variable $1 \leq z \leq K$. In addition, each model consists of logarithmic topic-mixture proportions $\vec{\theta}_{\tau,d}^i$ and document probabilities $\delta_{\tau,d}$ for each document $d \in \vec{D}_\tau^i$.

TopicTable summarizes the contents of documents by K document prototypes, which are derived from the K PLSA topics. TopicTable visualizes the relative strength of the k^{th} prototype in batch $\vec{D}_\tau^i$. This strength is equal to the probability $P(z = k|\zeta_\tau^i)$ of the k^{th} topic from which the k^{th} prototype was derived. This topic probability can be computed from the model parameters as follows.

$$
\begin{aligned}
P(z = k|\zeta_\tau^i) &= \sum_{d \in \vec{D}_\tau^i} P(z = k, d|\zeta_\tau^i) && (5.6)\\
&= \sum_{d \in \vec{D}_\tau^i} P(z = k|d, \zeta_\tau^i) P(d|\zeta_\tau^i) && (5.7)\\
&= \sum_{d \in \vec{D}_\tau^i} \exp(\theta_{\tau,d,k}^i) \exp(\delta_{\tau,d}^i) && (5.8)
\end{aligned}
$$

Each document of $\vec{D}_\tau^i$ has been annotated with tag τ by a particular user. The document IDs of batch $\vec{D}_\tau^i$ that have been annotated by user u are denoted by $\vec{D}_\tau^i(u)$. The influence of user u on the k^{th} topic is measured by the fraction of the topic probability which is due to the documents $\vec{D}_\tau^i(u)$. This fraction $g_{\tau,u,k}^i$ is derived as follows.

$$
b_{\tau,u,k}^i = \frac{\sum_{d \in \vec{D}_\tau^i(u)} P(z = k, d|\zeta_\tau^i)}{P(z = k|\zeta_\tau^i)} \qquad (5.9)
$$

Being fractions, these user influences $b_{\tau,u,k}^i$ with $1 \leq i \leq \bar{N}_\tau$, and $1 \leq k \leq K$ lie between 0 and 1.

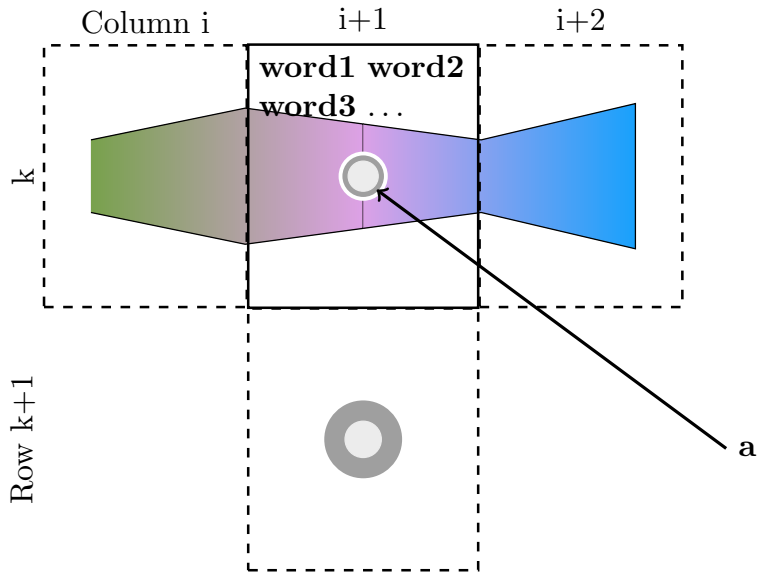


Figure 5.8: Each cell corresponds to one document prototype as described in the caption of Figure 4.3. Here, TopicTable is extended such that it visualizes the influence of a particular user u on the topics over time. **a)** To this end, the dark-gray circle, which visualizes the relative strength of a topic, is superimposed by a light-gray circle, which visualizes the influence of the user u.

5.4.2 Extending TopicTable for visualization of user influence

TopicTable visualizes the relative strength of a document prototype with a dark-gray circle whose size is relative to the probability of the corresponding topic (see Figure 4.3). In order to visualize a user's influence on a topic, these dark-gray circles are superimposed by a second light-gray circle as shown in Figure 5.8. The relation in size between the light-gray and the dark-gray circle roughly encodes the user influence on the corresponding topic. If the user is responsible for all contents which are represented by a topic, then the light-gray circle has the same size as the dark-gray circle. If the user has contributed none of the contents, then the light-gray circle vanishes.

The size of the light-gray circle for the k^{th} topic of the i^{th} batch is proportional to the probability fraction $P(z = k|\zeta_\tau^i) \cdot b_{\tau,u,k}^i$. Taking this probability fraction instead of the whole topic probability, TopicTable derives the size of the light-gray circle by the same equation which it uses for determining the size of the dark-gray circle. In more detail, TopicTable determines the radius of the light-gray circle as $(P(z=k|\zeta_\tau^i)\cdot b_{\tau,u,k}^i)^{5/7}/\sqrt{(2\pi)}$ [64].

Computing the size of the light-gray circle in this way gives the desired visualization effects. If the documents $\vec{D}_\tau^i(u)$ that have been tagged by user u contribute to the topic only to a minor degree, then the corresponding light-gray circle will be small and the dark-gray circle will dominate. Contrary, if a topic was defined by the documents of user u to a high degree, then the light-gray circle will be almost as large as the dark-gray circle. As a result, the light-gray circle will almost cover the dark-gray background circle. By inspecting the light-gray and dark-gray circles, a user might, at a glance, deduce relative strengths of the topics and how much these topics were influenced by the own documents under the tag in question.

Table 5.5: Active users for tag *learning* of the Bibsonomy system.

User	# Docs	first tagging	last tagging
69	301	2006-03-17	2008-12-28
2732	154	2008-06-19	2008-06-19
444	22	2006-10-27	2006-10-27
1771	20	2007-12-16	2007-12-16
3786	14	2008-12-27	2008-12-27

5.4.3 Case study

The extended TopicTable was applied to the Bibsonomy data. As an example, the semantic miscellany of the tag *learning* was studied as this tag is popular and, hence, is likely interpreted in different ways. Table 5.5 lists the five users who have annotated the most documents with the tag *learning*. In this example, the influence of user 69 on the learned PLSA topics over time is visualized. This user was chosen for the following two reasons. The user 69 clearly has tagged the most documents with the tag *learning*, and this user has been active almost throughout the whole studied period of time.

A sliding window of length 200 documents was applied to the stream of documents under the tag *learning* and it was shifted by 75% of its size. The last batch, which contained less then 200 documents, was omitted. This procedure resulted in four batches and successive batches overlapped by 50 documents. Each batch is annotated with the time stamp of its latest document.

Then, AdaptivePLSA was applied to these batches to learn four PLSA models with $K = 5$ topics over time. Five topics were chosen to obtain a coarse overview of the contents in the stream of documents under the tag *learning*. The hyper-parameters of the PLSA models were set to $\alpha = 1$ and $\beta = 1$. The result of applying the extended TopicTable to the learned PLSA models is presented in Figure 5.9. The determined user influences of user $u = 69$ are listed in Table 5.6.

The TopicTable in Figure 5.9 reveals the following thematic subjects of documents which have been tagged with the tag *learning*.

Topic 5

- from 2007-05 to 2008-06: cognitive process of learning; learning math; studies on influence of environment on learning activity; education of teachers; different ways of learning (via games, or in communities)
- in 2008-08: strong turn to thematic subjects on machine learning (*process*, *model*, *learner*, *image*, *technology*)

Topic 4 this topic changes relatively often as visualized by the small width of its background river

- in 2007-05: ABLA (Assessment of Basic Learning Abilities)
- from 2008-03 to 2008-6: machine learning; generative learning of statistical models; models for analyzing programming code
- in 2008-08: learning models (*learn*, *gener*) with genetic programming (*gp*, *function*, *evolv*, *fit*, *genet*)

Topic 3

- in 2007-05: learning and education via Web 2.0 (wikis)

• in 2008-03: machine learning; approaches and algorithms for classification problems; generative models; automata; feature selection
• from 2008-06 to 2008-08: emerging thematic subject on genetic programming (*gp*, *genet*, *perform*); learning models for classification (*classify*); learning in context of the semantic web (techniques and methods)

Topic 2
• in 2007-05: influences of playing video and computer games on learning abilities with respect to learning languages, and cognitive abilities
• from 2008-03 to 2008-06: cultural aspects of learning; social learning in context of pedagogy (*socy*, *learn*, *theory*, *imitat*, *dynam*); studying the cognitive process of learning by measuring neural activities; social networks (gaining knowledge out of modeling social networks)
• in 2008-08: reinforcement learning (*agent*, *robot*, *interact*, *complex*, *algorithm*); evolutionary learning

Topic 1
• from 2007-05 to 2008-06: value of computer applications for learning; design and development of computer games for learning; collaborative filtering
• in 2008-08: developing and designing software (in context of machine learning)

These findings give clues to several thematic subjects that users of Bibsonomy associate with the tag *learning*. Two main thematic subjects are learning as a cognitive process and learning in the context of machine learning. Some thematic subjects under the tag *learning* change with time. For instance, the subject on machine learning with the help of genetic programming emerges in 2008-06 whereas the subject on ABLA is present in 2007-05 and quickly vanishes thereafter.

Inspecting the gray circles in the middle of each box of the TopicTable, one finds that the first and the fifth topic are the dominating topics from the beginning until 2008-06. Later, in 2008-08, the third topic becomes the strongest one.

The light-gray circles on top of the dark-gray circles visualize the influence of user 69 on the learned topics. The documents which this user has tagged with the tag *learning* strongly influence the first and fifth topic from the beginning until 2008-06. In 2007-05, the documents of user 69 have a strong influence on the second and third topic. In the last column 2008-08, user 69 seems to be responsible for less documents under the tag *learning*. The user has an influence on the first and fifth topic only to a minor degree. This becomes especially obvious when inspecting Table 5.6. Interestingly, the strength of the first and fifth topic decreases from 2008-06 to 2008-08 as the gray circles become smaller. At the same time, the background rivers of these two topics become narrower indicating a thematic shift of the underlying topics. All these observations lead to the hypotheses that (i) user 69 mainly influenced the first and fifth topic, that (ii) the thematic subjects which user $u = 69$ links with the tag *learning* diminish in 2008-08 when the user 69 is less active, and (iii) that the first and fifth topic start to focus on different thematic subjects when the user $u = 69$ is less active. Last, because of the user's strong influence on the first and fifth topics until 2008-06, it might be hypothesized that this user mainly uses the tag *learning* for tagging documents about learning as a cognitive process.

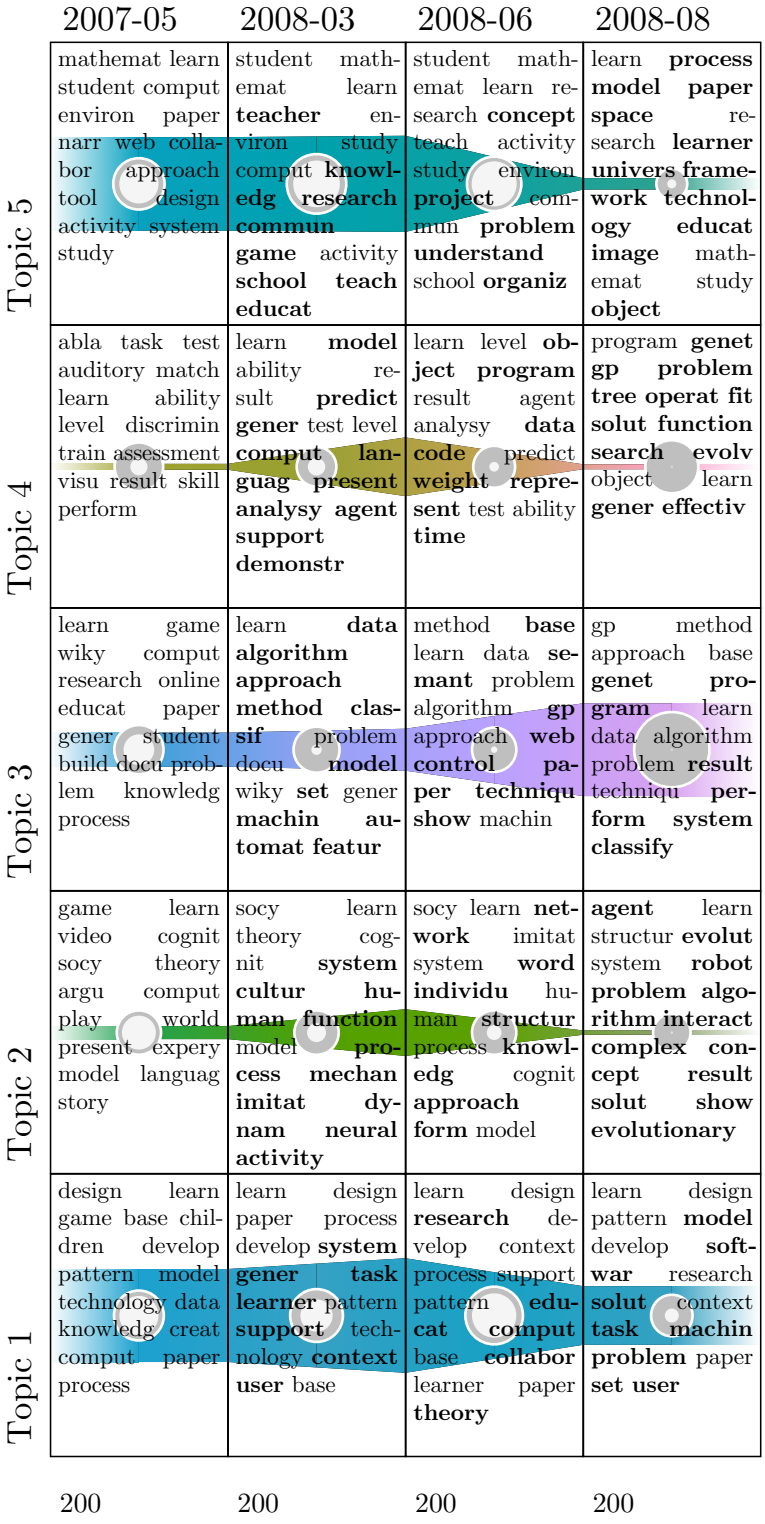


Figure 5.9: Extended TopicTable for tag *learning* from the Bibsonomy data. Document prototypes consist of the 15 most likely words for each topics. Light-gray circles on top of dark-gray circles visualize how strongly the underlying topics are influenced by documents user 69 has tagged with the tag *learning*.

Table 5.6: The influence of user 69 on social tag *learning* as measured by $b^i_{\tau,u,k}$ (values in %).

Topic k	**2007-05 (i=1)**	**2008-03 (i=2)**	**2008-06 (i=3)**	**2008-08 (i=4)**
5	73	75	82	17
4	23	33	12	0
3	54	11	5	0
2	74	31	21	0
1	76	53	73	21

5.5 Conclusions and future directions

Recently emerged collaborative tagging systems are part of the growing Social Web. Users of these systems manage and share own resources such as links to music, documents, and images. Beside managing, users organize resources by assigning tags to them. Often, any short phrase or word might be used as a tag since tags are not centrally supervised.

Users and researchers became aware of the risk of semantically overloaded tags which are a consequence of the many different conceptions that users associate with the same tag. Semantic overloaded tags could reduce the effectiveness of organizing resources by tagging. On the other hand, they are of great value as getting to know which thematic subjects are associated with a tag is learning about the conceptions of this tag and thematic subjects of resources this tag is linked to. Learning about thematic subjects a tag is associated with is making sense out of the many resources this tag is assigned to. For systems that deal with document-like data, two methods for investigating the diversity of meanings a tag is associated with are proposed in this chapter. Both methods complement each other; the first method sheds light on minor meanings of social tags whereas the second method visually summarizes contents of documents under a tag for deducing main thematic subjects this tag is associated with.

The semantic miscellany of social tags of the Bibsonomy and CiteULike system was investigated in two case studies. Users of the Bibsonomy system, for example, associate the tag *learning* with different thematic subjects such as machine learning and learning as a cognitive process. Moreover, both proposed methods assist the reader in discovering changes of the semantic miscellany of tags with time. For instance, users of the CiteULike system associate with the tag *human* minor thematic subjects over time as different as the human auditory system, human ontogenesis and phylogenesis, and human odor from social and psychological perspectives.

Both proposed methods might be effective tools for users who want to learn about the semantic miscellany of social tags. As such, the proposed methods might complement existing approaches for tag sensemaking such as tag clouds, collocate clouds or links from tags to WordNet. The potential of methods for clarifying the many meanings of social tags is twofold. First, they make the work with collaborative tagging systems more effective. Second, they enable users to gain knowledge from the "wisdom of the many". Learning as a social process often is learning from the others. In context of collaborative tagging systems, learning from the others is learning what other users associate with particular tags. By that, informed users might be able to understand better which particular thematic subject of a document is expressed by a certain tag.

The development of sophisticated methods for tag sensemaking is important as the

Social Web will continually grow in the future. One direction of enhancing the proposed method for detecting minor meanings under a tag is to remove the following strong assumption: a user has one unique interpretation of a tag. So far, this assumption is implicitly made by the proposed approach but, obviously, this strong assumption might be often missed as a user might associate different meanings with the same tag. Removing this assumption means to explicitly take into account different conceptions that a user might have about a tag. A second future direction is to take into account the size of the set of number of documents which should be investigated for contributing a minor meaning. The characteristic distances of documents to the mean representation might be different with respect to the size of the investigated document sets. For small document sets, observed distances might be larger and more disperse than for larger document sets. These differences could be taken into account, e.g., by a bootstrap procedure which is specific for the size of the document set that should be investigated. To this end, the proposed bootstrap procedure could be restricted to draw only bootstrap users which have contributed similarly many documents as the size of the investigated document set. Size-specific bootstrap procedures could reduce the number of falsely detected or missed minor meanings and thereby enhance the detection of minor meanings.

TopicTable could be further extended for visual sensemaking in the following ways. A close relation between users and tags of collaborative tagging systems exists: users define tags and get inspired by tags of other users. Hence, in addition to analyzing topics under tags, learning and presenting topics under users could be another tool for tag sensemaking. Knowing which user is responsible for which topics under a tag and, at the same time, inspecting topics under this user might provide additional information about the semantic miscellany of the tag under inspection. Another extension could be to take common topics under several tags into consideration. So far, the combination of AdaptivePLSA and TopicTable learns and visualizes topics under a tag independently of topics under the other tags. Discovering and presenting topics that are common under several tags will greatly enhance studying the semantic miscellany of tags by shedding light on the semantic relationship among tags.

Chapter 6

Bilingual topic modeling of chemical compounds

Drug discovery is part of natural-product chemistry, an interdisciplinary field between chemistry and biochemistry with the subject of searching for yet unknown active natural products for developing new drug leads. Natural products are chemical compounds synthesized by living organisms. Examples are glucose, ethanol and paclitaxel, which are, for instance, synthesized by plants, baker's yeast and the pacific yew tree, respectively. After isolation of a promising natural product, researchers investigate this chemical compound. Beside other investigations, the *chemical constitution* of the isolated chemical compound is of great interest. The chemical constitution describes (i) which atoms constitute the compound, and (ii) how these atoms are connected by which chemical bonds. The chemical constitution of a compound is often represented by drawing a structural formula as shown in Figure 6.1 for toluene. The process of determining the chemical constitution and the three-dimensional structure of a chemical compound is called structure elucidation. In this chapter, bilingual topic modeling of chemical compounds is proposed. Applications thereof might give valuable hints, for example, for structure elucidation of new chemical compounds.

For investigating the chemical constitution a variety of experimental methods is available. In combination with other experimental procedures *Nuclear Magnetic Resonance* (NMR) spectroscopy, by which researchers might draw conclusions about how atoms are connected by which chemical bonds, is often applied. NMR refers to the ability of specific atom nuclei to interact with an electromagnetic field. For a given chemical compound, NMR spectroscopy records these interactions and visualizes them as peaks in a NMR spectrum. An example of a 2D NMR spectrum of toluene is shown in Figure 6.1. The interaction of a single atom and, hence, the position, shape, and intensity of the corresponding NMR peaks are affected by different factors. For example, the chemical bonds in which this atom participates affects the quality of interaction. Another factor is the neighbor atoms which are connected to this atom via chemical bonds.

For structure elucidation researchers study NMR spectra and draw conclusions about the chemical constitution from peak positions, shapes, and intensities. Structure elucidation is challenging and laborious as researcher use a combination of different NMR experiments and other complementing experimental procedures to work out hypotheses about the chemical constitution of structural fragments of the studied chemical compound. These hypotheses are combined, piece by piece, and the combinations have to

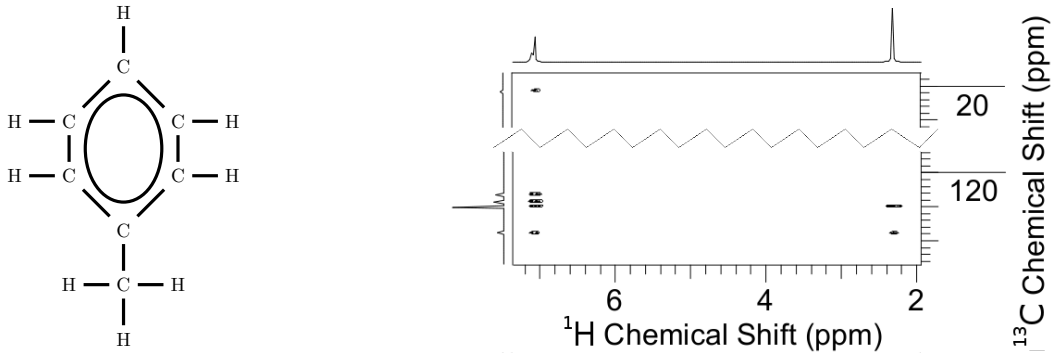


Figure 6.1: Structural formula and 2D NMR spectrum of toluene. Peaks are projected onto the two-dimensional plane. Peak positions are defined in terms of (chemical) shifts, whose measurement unit is parts per million (ppm), of the peak with respect to reference peaks.

be verified by additional experiments to eventually lead to a final structure suggestion.

In this work, bilingual topic modeling of chemical compounds is proposed as help for structure elucidation. To this end, chemical compounds are described in two different languages. The first language describes their chemical constitutions, whereas the second language describes positions of NMR peaks in their NMR spectra. Describing chemical compounds with help of these two languages, bilingual topic modeling of chemical compounds makes possible (i) to predict distributions over peak positions from a known chemical constitution, and (ii) to predict distributions over structural fragments from a given NMR spectrum. Fingerprints which consist of the most likely predicted NMR signals or structural fragments might be derived from these distributions. Applications of such fingerprints could be (i) to strengthen or alleviate structural hypotheses during the process of structure elucidation, and (ii) to use these for information retrieval and for look-ups in chemical databases. For example, if a researcher is investigating the chemical constitution of a new compound using NMR, then the researcher might want to search in structure databases for structurally most similar compounds. In this situation, the researcher could predict structural fragments from the NMR spectrum by bilingual topic modeling and use these fragments for the database search. In the opposite direction, a researcher could predict peak positions from a known chemical constitution and use these predicted positions for search in NMR databases.

It should be emphasized here that this work on bilingual topic modeling of chemical compounds does not aim at competing with expert systems for structure elucidation of chemical compounds. As structure elucidation is a complex problem, expert systems combine a variety of data from different experimental methods and predict a few structure suggestions.

The rest of this chapter is organized as follows. Next, related work is presented in Section 6.1. In Section 6.2, the two languages used for describing chemical compounds are described in detail. The utilized multilingual topic model and the learning approach are presented in Section 6.3. The experiments and discussions of the results are subject of Section 6.5. Last, conclusions and future directions are given in the Section 6.6.

6.1 Related work

Different approaches for the prediction of NMR peak positions from the chemical constitution of chemical compounds have been proposed. Such approaches are specific for the type of NMR signals that should be predicted, e.g., for ^{1}H, ^{13}C, ^{15}N or ^{31}P. In addition, they are specific for the different NMR experiments, e.g., 1D and the many different 2D NMR experiments. A diversity of computational approaches have been proposed for the prediction of NMR peak positions. For ^{13}C 1D NMR this diversity is reviewed in the introduction of the work [89] and it includes empirical methods like linear and multivariate regression, artificial neural networks, incremental models, genetic programming approaches, and combinations thereof, as well as quantum-mechanical methods [90]. A comparison of empirical and quantum-mechanical methods is given in the work [91]. All these methods for the prediction of NMR peak positions use information about the atoms and chemical bonds of a chemical compound as input. Different approaches differ by the way how they encode this information by descriptors. Examples are descriptors for steric and electronic properties derived by the Hückel method [92], functional group descriptors [93], topological descriptions of magnetic atoms as described in [94], atom-centered topological, geometric and electronic descriptors as selected by genetic programming [95], and descriptors based on atomic distance-edge vectors [96]. Beside this diversity, approaches for the prediction of NMR peak positions differ with respect to different research goals. For example, prediction of chemical shifts could be done for the structure elucidation of one or a few compounds of interest like in [97] where the chemical constitution of two germacrane derivatives was elucidated by a tight integration of shift prediction and experimental methods. Other approaches predict NMR peak positions for members of a specific class of chemical compounds like keto-steroids [98] or monosaccharides [99]. Yet another class of approaches predict NMR peak positions for a broader class of more diverse chemical compounds like natural compounds. Such approaches are often part of commercial software for analyzing NMR data like ACD[1] or ChemWindow[2].

Multilingual topic modeling, which was originally proposed for modeling multilingual document corpora, is adapted to modeling chemical compounds in this chapter. Beside predicting positions of NMR peaks, the here proposed approach might predict also structural fragments from a given NMR spectrum. Approaches for structure prediction might be categorized into those which propose structural fragments and others that predict complete chemical constitutions. The latter approaches rely on a combination of different kinds of data such as 1D and 2D NMR spectra. All these data are combined and used by systems for computer-aided structure elucidation (CASE) which derive hypotheses about the complete chemical constitution of a studied chemical compound. CASE systems are specific for a class of chemical compounds. Examples include the prediction of the tertiary structure of proteins in the field of proteomics and the prediction of the chemical constitution of small chemical compounds in the field of metabolomics. CASE systems for small organic molecules, which rely on 1D and 2D NMR data, are reviewed by Elyashberg et al. [100]. Examples include StrucEluc [101], the variants of SESAMI [102, 103], SENECA [104, 105], and the LSD system [106].

Other approaches determine hypotheses about structural fragments instead of coming

[1] `acdlabs.com`, May 8, 2012
[2] `bio-rad.com`, May 8, 2012

up with suggestions of the complete chemical constitution. Hypotheses about structural fragments are useful, e.g., for the validation of complete structure suggestions. Often, prior knowledge about peak positions of specific NMR experiments is taken into account. An example is the application of a Bayesian network [107] to the prediction of structural fragments from ^{13}C NMR peaks where the graph structure of the Bayesian network was specifically tailored to ^{13}C NMR peak positions of benzene derivatives.

The bilingual topic model, which is used in this work, was originally proposed in the field of multilingual topic modeling of multilingual document corpora. Approaches for multilingual topic modeling could be categorized into those approaches that rely on aligned documents and those that do not. Aligned documents are documents that share the same content and differ in their language. The protocols of the European parliament, which are written in different languages and share the same contents, are an example. For each protocol in one language there is a corresponding document in another language. Documents might be aligned to different degrees; they might be direct translations of each other on sentence or even on word level, or they might be about the same thematic subjects but are not translations of each other. Wikipedia articles which are about the same subject and which have been written independently of each other in different languages are an example of the latter kind of multilingual documents. In contrast, a corpus of unaligned documents is simply a collection of documents which are written in different languages and which do not share common thematic subjects.

The basic idea of multilingual topic models that rely on aligned documents is to exploit the information about which documents are translations of each other. An example is the statistical mixed membership model proposed by Stephen et al. [108]. They apply this model to PNAS[3] articles. The two languages are: words of the abstracts and references of the bibliographies. Hence, for each PNAS article Stephen et al. derive two corresponding documents: the abstract and the references. The central assumption for learning topics from these data is that corresponding documents have equal topic-mixture proportions. For each learned topic, Stephen et al. studied which are the articles and references that are most strongly connected to this topic. Mimno et al. [109] and Ni et al. [110] have extended LDA to the polylingual topic model[4] for inferring topics from aligned multilingual corpora. The proposed LDA extension models documents which are about the same subjects but do not have to be word-to-word translations of each other. For learning the information about the topic mixture is shared among the aligned documents: corresponding documents are assumed to have the same topic-mixture proportions. Exploiting this constraint, the polylingual topic learns realizations of the same underlying topic in different languages. Zhao et al. [111, 112] propose the bilingual topical admixture model (BiTAM) and its extension, the hidden Markov-BiTAM. These models rely on bilingual corpora of aligned documents and are capable of inferring word-to-word translations. To this end, the aligned documents have to be translations of each other and the models exploit the information about which sentences of aligned documents are direct translations of each other. Another approach for inferring word-to-word translations is taken by Tam et al. [113]. Again, the corresponding documents have to be direct translations of each other. Tam et al. learn a single topic model for each language and couple these models by assuming that topic-mixture proportions of corresponding

[3]Proceedings of the National Academy of Sciences

[4]Both groups came up with the same LDA extension at the same time. Mimno et al. call this extension polylingual topic model.

documents are equal to each other. Platt et al. [114] propose coupled PLSA for modeling corpora of aligned documents. In contrast to the polylingual topic model [109], which directly enforces aligned documents to have the same topic-mixture proportions, coupled PLSA assumes each of the corresponding documents to have own specific topic-mixture proportions. For inference Platt et al. regularize the posterior of the model [115, 116] in order to favor similar topic-mixture proportions for corresponding documents.

The popular assumption for aligned documents to have similar topic-mixture proportions is meaningless in case of unaligned documents. Instead, topic models for unaligned multilingual documents often rely on some kind of dictionary of word-to-word translations for coupling topics in different languages. Boyd-Graber et al. [117] have extended LDA for modeling multilingual corpora of unaligned documents. They define tuples of aligned words; each tuple contains translations of a certain word in each of the different languages of the document corpus. Different to LDA, a topic is not a distribution over words of one language but over such tuples. Words of each language without known translation are drawn from one language specific background topic. Tuples of word translations are partially predefined and learned from the data. Word translations are learned by assuming that the same word in different languages occurs together with the same context words, e.g., the word *Hund* should appear in German texts together with *Leine* and *Bellen*, and the translated English word *dog* should appear in English texts together with *leash* and *bark*. As a consequence of topics being distributions over tuples of word translations, a word and all of its translations have the same word probability for a given topic. A similar line of action is taken by Daumé et al. [118] who propose JointLDA, a generalization of the model proposed by Boyd-Graber et al. [117]. Differences are (i) JointLDA is capable of taking into account several translations of a word, and (ii) JointLDA uses multiple monolingual background distributions for each language with the help of which it offers a richer expressiveness. Zhang et al. [119] have extended PLSA to Probabilistic cross-language Latent Semantic Analysis (PCLSA). Similar to Boyd-Graber et al. [117], PCLSA does model unaligned documents. PCLSA uses a fixed dictionary of word translations for the purpose of determining words and their corresponding translations in documents which are written in different languages. This knowledge is exploited to learn topics that are coherent in different languages. In contrast to Boyd-Graber et al. [117], word probabilities of a word and its translations might vary among realizations of a topic in different languages. Because of this capability, PCLSA might better capture differences among realizations of one topic in different languages.

6.2 Two languages describing chemical compounds

The chemical compounds are described in two languages. The first language describes the chemical constitution of a chemical compound. Words of this language are structural fragments, which one can think of as building blocks of chemical constitutions. With the help of the second language, the positions of peaks of a 2D NMR spectrum are described. Both languages describe electromagnetic interactions among neighbor atoms of the chemical compound. The structure language does so by directly encoding neighbor atoms and electromagnetic properties. The NMR languages does so by describing positions of NMR peaks. The peak positions encode the electromagnetic interactions among

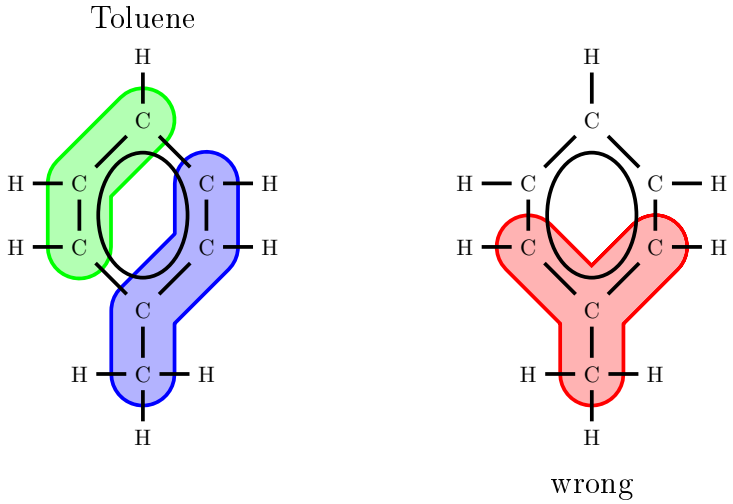


Figure 6.2: Atom sequences of length 2 (green) and 3 (blue) for the structural formula of toluene. The ellipse in the middle symbolizes aromatic π-bonds. The red subset of connected atoms is not an atom sequence as the participating atoms are not connected in a linear manner and the corresponding path would contain cycles.

neighbor atoms as measured by NMR spectroscopy. As a result, the two documents which are constructed from the same chemical compound correspond to each other.

6.2.1 Atom sequences

The chemical constitution of a chemical compound describes the number and kind of atoms of the compound, and how these atoms are connected via which chemical bonds. A visualization of the chemical constitution is the structural formula which displays atoms and chemical bonds in a two-dimensional plane. An example is shown in Figure 6.1 which depicts the structural formula for toluene.

Atom sequences [120] are linear, structural fragments which are derived from the structural formula. As such, the collection of atom sequences which have been derived from a structural formula encode local topological properties of the chemical constitution of a chemical compound.

For the purpose of defining atom sequences, structural formulas are described by an undirected graph $\mathcal{G} = (V, E)$ with node set V and edge set E. For each atom of the chemical compound the node set contains exactly one node which uniquely identifies this atom. For each chemical bond between two atoms which are identified by the nodes $v_i, v_j \in V$, $v_i \neq v_j$, the edge set contains exactly one edge $\{v_i, v_j\} \in E$. A path of length $d \geq 0$ in the graph $\mathcal{G}$ is a sequence of $d+1$ nodes $v_1, \ldots, v_{d+1}$ such that the following two constraints are fulfilled. All nodes are element of the node set: $v_i \in V$ with $1 \leq i \leq d+1$. Further on, there must be an edge $\{v_i, v_{i+1}\}$ in the edge set for each pair of successive nodes (v_i, v_{i+1})), $1 \leq i \leq d$, of the path.

The set of all atom sequences of length d which are derived from a structural formula as represented by graph $\mathcal{G}$ is the set of all paths in graph $\mathcal{G}$ of length d without cycles. The essential part of this definition is that the paths must not contain cycles. Two atom

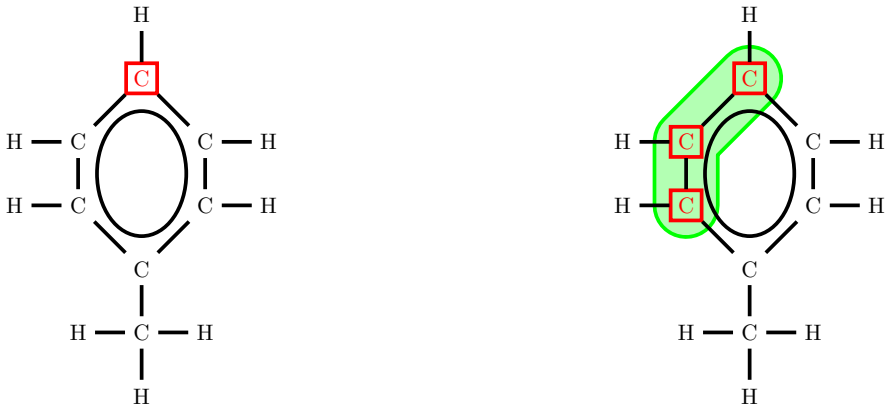


Figure 6.3: The three atom descriptors take the same values $(6, 2, 1)$ for all highlighted carbon atoms. The atom sequence highlighted in green is described by the word $(6, 2, 1)|(6, 2, 1)|(6, 2, 1)$.

sequences of length 2 (green) and 3 (blue) are shown in Figure 6.2. Atom sequences[5] were proposed by Nilakantan et al. [120] who defined atom sequences only of length 3. The definition which is given in this chapter is slightly more general as it defines atom sequences of arbitrary length $d \geq 0$.

For constructing an atom-sequence document from a given chemical compound, first, all atom sequences are determined from the structural formula of this compound. Then, each atom sequence is mapped to one word which describes its electromagnetic characteristics and which is put into the document. This word is constructed by describing each atom of the atom sequence by the following three atom descriptors that partially capture their electromagnetic properties.

1. the atomic number, which encodes the atom type, e.g., carbon, oxygen
2. the number of direct neighbor atoms that are different from protons (H)
3. the number of π electrons, which encodes in how many multiple bonds the atom participates

For example, the atomic number of the carbon atom highlighted in Figure 6.3 (left) is 6. This carbon atom has two direct, non-proton neighbors and one π electron as this carbon atom participates in the aromatic bond. Thus, the atom descriptors take the values $(6, 2, 1)$. Aligning the values of the three atom descriptors for all atoms of an atom sequence eventually gives the word that represents this atom sequence. For instance, the green atom sequence in Figure 6.3 (right) is represented by the word $(6, 2, 1)|(6, 2, 1)|(6, 2, 1)$.

A fourth useful information about an atom is the number of its direct neighbor protons. This information is implicitly captured by the three descriptors. The reason is that the sum of the numbers of direct, non-proton neighbors and π electrons is fixed at a value which is specific for each kind of atom. For example, for carbon atoms this value is 4. Hence, knowing the atom number, the number of direct, non-proton neighbors and the number of π electrons determines the number of direct neighbor protons. For example,

[5] Nilakantan et al. use a term different from the term *atom sequence*.

the carbon atom which is highlighted in Figure 6.3 (left) has two non-proton neighbors and one π electron. Hence, this carbon atom has one $(4 - 2 - 1 = 1)$ direct neighbor proton.

Following Nilakantan et al. [120], atom sequences are derived after having neglected all protons from the structural formula. These protons are still represented by the atom descriptors of the other atoms. An advantage of omitting protons is that the number of possible atom sequences, i.e., the vocabulary of the atom-sequence language, is reduced. But in contrast to Nilakantan et al., who additionally neglected some possible atom sequences, in this work all possible atom sequences are derived once the protons have been omitted from the structural formula.

6.2.2 2D NMR

The second language encodes positions and the number of peaks of NMR spectra. NMR refers to the ability of magnetic atom nuclei such as ^{1}H, ^{13}C, or ^{15}N to absorb electromagnetic energy in a magnetic field. In 1D NMR spectroscopy, an electromagnetic pulse which is specific for one kind of magnetic target nuclei is applied to a probe that contains the chemical compound for which a NMR spectrum should be recorded. After applying the pulse, the absorbed energy is radiated from each target nucleus at a specific resonance frequency which is recorded by the NMR spectroscope. Eventually, each recorded frequency is visualized as a peak in a 1D NMR spectrum, where the position and the height of the peak corresponds to the frequency and its amplitude. Peak positions are measured in relation to a reference frequency, and so are the coordinates of NMR peaks called *chemical shifts*. The resonance frequency, and so the peak position, depends on the kind of target nucleus. Often, 1D NMR experiments are specific for protons and the recorded spectra are ^{1}H 1D NMR spectra, in short ^{1}H NMR. The resonance frequency depends on the strength of the local magnetic field at the target nucleus, too. This local magnetic field is influenced by the chemical bonds, in which the atom participates, and other factors like magnetic interactions to neighbor nuclei and shielding effects of the electromagnetic neighborhood. In summary, the resulting 1D NMR spectrum contains peaks of all target nuclei, e.g., protons, of a studied chemical compound. Peak positions encode structural information about the target nuclei, e.g., chemical bonds they participate in and neighbor nuclei.

Often, 1D NMR spectra are crowded by many peaks as such spectra contain peaks for all target nuclei of a chemical compound. As a result, some peaks may overlap each other and thereby reduce the resolution of 1D NMR spectroscopy. A solution to this problem is 2D NMR spectroscopy. Peaks, which lie close to each other in a 1D NMR spectrum, might be separated by the additional dimension. A further advantage of 2D NMR spectroscopy is the capability of specifically detecting magnetic interactions among target nuclei of two different kinds. For example, with the help of ^{1}H/^{13}C NMR spectroscopy, researchers might specifically record interactions among direct protons and carbon atoms which are connected via one (heteronuclear single quantum correlation spectroscopy, HSQC) or multiple chemical bonds (heteronuclear multiple bond correlation spectroscopy, HMBC). Such interactions are a useful source of information about local structural properties, e.g., groups of connected atoms of an investigated chemical compound. Consequently, 2D NMR spectroscopy, and in particular ^{1}H/^{13}C NMR, is of great value for structure elucidation of chemical compounds.

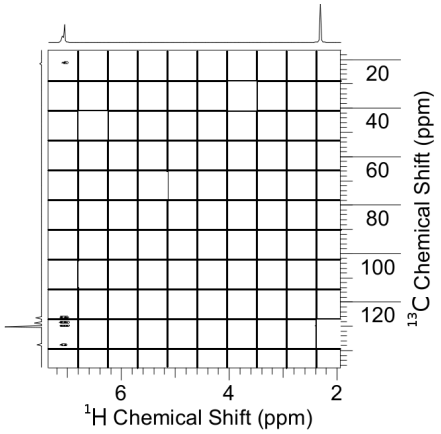


Figure 6.4: Discretized ^{1}H ^{13}C HMBC spectrum of toluene. Each bin corresponds to exactly one word of the HMBC (2D-NMR) language.

HMBC spectroscopy is such a 2D NMR experiment for detecting electromagnetic interactions among nuclei of two different types. By ^{1}H/^{13}C HMBC spectroscopy, chemists are able to detect interactions among protons and ^{13}C carbon nuclei through multiple (two up to four) chemical bonds. Detected interactions are represented by peaks of a two-dimensional ^{1}H/^{13}C HMBC spectrum. The ^{1}H/^{13}C coordinate of a peak is specific for the proton/carbon which participates in such an interaction. These coordinates are affected by the electromagnetic neighborhood of the interacting protons and ^{13}C carbon nuclei. From a ^{1}H/^{13}C HMBC spectrum, researchers might draw conclusions about which protons of a chemical compound can interact with neighbor (via two up to four chemical bonds) carbon atoms. Figure 6.1 shows an example of a ^{1}H/^{13}C HMBC spectrum for toluene. In summary and important for this work: ^{1}H/^{13}C HMBC spectroscopy maps local interactions among protons and carbon atoms of a chemical compound to positions of peaks of a two-dimensional ^{1}H/^{13}C HMBC spectrum. This type of 2D NMR spectra capture useful structural information about groups of connected protons and carbons. More details on NMR spectroscopy can be found elsewhere [121–123]. To keep notation uncluttered, ^{1}H/^{13}C HMBC is shortly called HMBC in the rest of this work.

The second language for the description of chemical compounds describes peak positions of HMBC spectra. In order to derive words from peak positions of such a two-dimensional spectrum, this spectrum is discretized into regular two-dimensional bins which are two-dimensional intervals of ^{1}H and ^{13}C chemical shifts. Each bin becomes one word of the HMBC vocabulary. An example of a discretized HMBC spectrum is shown in Figure 6.4. For a given HMBC spectrum, the HMBC document is constructed as follows. For each peak which falls into a two-dimensional HMBC bin the corresponding HMBC word is put into the document. Consequently, each word occurs as often in the document as peaks fall into the corresponding HMBC bin.

In principle, NMR descriptions of chemical compounds might be deduced from any kind of NMR experiment. HMBC spectra are used as an example in this work for the reason that these 2D NMR spectra encode information about electromagnetic interactions of protons and carbons which are connected via multiple chemical bonds. This kind of information is similar to the kind of electromagnetic information about a few linearly connected atoms as encoded by atom sequences. Hence, the two languages of atom-

sequences and HMBC bins seem to be particularly well suited for describing chemical compounds in order to model these by bilingual topic model.

6.3 Polylingual topic model

The polylingual topic model as introduced by Mimno et al. and Ni et al. [109, 110] is used in this work. This model learns word distributions over documents which are written in L different languages. Documents about the same subject but written in different languages are called corresponding documents. Corresponding documents must be about the same subjects but do not have to be word-to-word translations of each other.

A polylingual topic model consists of K polylingual topics. Each polylingual topic has L realizations each of which in one of the L languages. When a word of a document is drawn from a polylingual topic then this word is drawn from that topic realization which corresponds to the language of the document. The following fundamental assumption underlies the polylingual topic model. All documents of a tuple of corresponding documents have the same topic-mixture proportions. This assumption is trivially fulfilled if the corresponding documents are translations of each other. In this case the thematic subjects which are present in one document obviously will be present in the other corresponding documents, too.

In this work, the polylingual topic model is applied to representations of chemical compounds in two languages. These representations could be understood as documents whose languages are atom sequences and HMBC signals. As discussed earlier, the two corresponding documents describe similar electromagnetic interactions among atoms of a chemical compound. Whenever a pattern of such interactions is present in the atom-sequence document then this pattern is likely present in the HMBC document as well. Since two languages are used in this work, the topic models and topics are also called bilingual topic model and bilingual topics. The learned bilingual topics capture patterns of interactions which are described in the two domains of atom sequences and of HMBC signals. A devised example of one bilingual topic is shown in Figure 6.5 where the most likely structural fragments as described by atom sequences are shown and the most likely two-dimensional NMR bins of a HMBC spectrum are highlighted.

Next, the representation of the data is explained in Section 6.3.1. Afterwards, in Sections 6.3.2 and 6.3.3, the polylingual topic model and its prior are described. Then the Expectation Maximization algorithm for parameter learning is derived in Section 6.3.4.

6.3.1 Data representation and notation

The L languages are indexed by a language ID $l \in \{1, \ldots, L\}$. The document corpus consist of tuples of L corresponding documents, each of which is written in one of the L languages. A tuple of L corresponding documents is called a polylingual document. Each polylingual document is identified by one document ID $d \in \{1, \ldots, N\}$.

The vocabulary of the l^{th} language consists of M_l words and so are the words of this language enumerated by word IDs $1 \leq w \leq M_l$. A pair of a word ID and a language ID (w, l) uniquely identifies the word w in language l. The same word ID in combination with different language IDs might refer to different words in different languages.

Documents are assumed to be bag of words, i.e., the ordering of words is neglected.

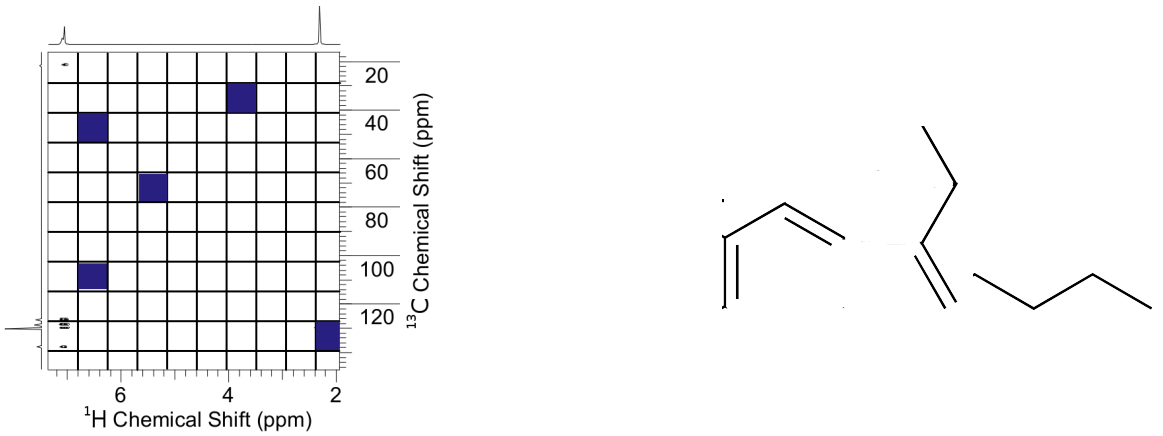


Figure 6.5: A bilingual topic has realizations in both languages: atom sequences and 2 NMR bins of a HMBC spectrum. Highlighted and listed are the most likely 2D NMR bins and structural fragments as described by atom sequences of a devised bilingual topic.

Consequently, a triple (d, w, l) of a document ID, a word ID and a language ID is added to the data $\vec{X}$ for each occurrence of a word with ID w from language l in polylingual document d. The number of triples in data $\vec{X}$ is denoted by $|\vec{X}|$. This number is equal to the number of all word occurrences in all documents across all languages.

6.3.2 Likelihood model

In this chapter, the polylingual topic model is defined in a slightly different manner as was done by Mimno et al. and Ni et al. [109, 110]. In more detail, the polylingual topic model is extended by language IDs.

A plate model representation of the polylingual topic model is depicted in Figure 6.6. After the model parameters haven been drawn from their priors, the data triples $(d, w, l)_i$ with $1 \leq i \leq |\vec{X}|$ are sampled from a polylingual topic model with K polylingual topics as follows.

1. sample document ID d_i from discrete distribution over all document IDs

$$d_i \sim P(d)$$

2. sample a topic index $1 \leq z_i \leq K$ from the topic-mixture proportions $P(z|d_i)$ of document d_i

$$z_i \sim P(z|d_i)$$

3. sample a language ID $1 \leq l_i \leq L$ from discrete distribution $P(l|d_i)$ given document d_i

$$l_i \sim P(l|d_i)$$

4. sample a word $1 \leq w \leq M_{l_i}$ from the realization of the z_i^{th} topic in the l_i^{th} language

$$w_i \sim P(w|z_i = k, l_i)$$

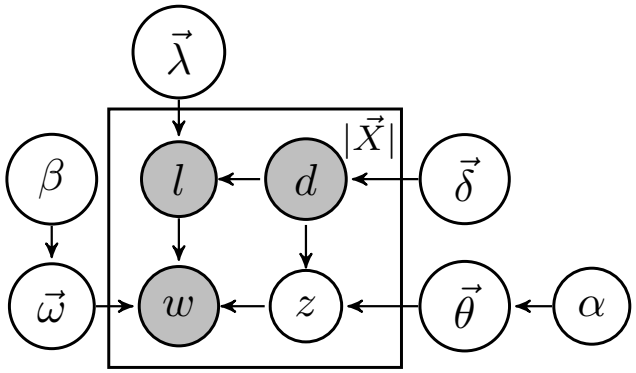


Figure 6.6: Plate model representation of polylingual topic model.

All discrete probability distributions are generalized Bernoulli distributions.

The parameters of these generalized Bernoulli distributions constitute the parameter set of a polylingual topic model ζ. In the following these parameters are defined with respect to document, topic and language IDs, which lie within the ranges $1 \leq d \leq N$, $1 \leq k \leq K$, $1 \leq l \leq L$, respectively.

1. logarithmic document probabilities $\vec{\delta} = (\delta_d)_{1 \leq d \leq N}$ with $\delta_d = \log P(d)$
2. logarithmic topic-mixture proportions for each document $\vec{\theta}$ which are defined as $\vec{\theta} = (\vec{\theta}_d)_{1 \leq d \leq N}$ with $\theta_{d,k} = \log P(z = k|d)$
3. logarithmic language-mixture proportions for each document $\vec{\lambda} = (\vec{\lambda}_d)_{1 \leq d \leq N}$ with $\lambda_{d,l} = \log P(l|d)$
4. logarithmic word-topic associations $\vec{\omega} = (\vec{\omega}_{k,l})_{1 \leq k \leq K, 1 \leq l \leq L}$, the l^{th} realization of the k^{th} polylingual topic is $\vec{\omega}_{k,l} = (\omega_{k,l,w})_{1 \leq w \leq M_l}$ with $\omega_{w,k,l} = \log P(w|z = k, l)$

The motivation for using logarithmic parameters is that these correspond to the natural parametrization of the generalized Bernoulli distributions as discussed in more detail in Section 2.5.

The likelihood, which is marginalized over the unobserved topic-index variables as given by $\vec{Z} = (z_i)_{1 \leq i \leq |\vec{X}|}$, reads as follows.

$$P(\vec{X}|\zeta) = \prod_{i=1}^{|\vec{X}|} P((d,w,l)_i|\zeta) \tag{6.1}$$

$$P((d,w,l)_i|\zeta) = \sum_{z_i=1}^{K} P((d,w,l)_i, z_i|\zeta) \tag{6.2}$$

$$= \sum_{z_i=1}^{K} P(w_i|d_i,l_i,z_i,\zeta)P(l_i|d_i,z_i,\zeta)P(z_i|d_i,\zeta)P(d_i|\zeta) \tag{6.3}$$

Given the model parameters, the following two conditional independence assumptions hold due to the definition of the generative process: (i) the probability of a word given its language ID and topic ID is independent of the document ID, mathematically expressed as $P(w_i|d_i,l_i,z_i,\zeta) = P(w_i|l_i,z_i,\zeta)$, and (ii) the probability of the language ID

given the document ID is independent of the topic index, i.e., $P(l_i|d_i, z_i, \zeta) = P(l_i|d_i, \zeta)$. Exploiting these independence assumptions, we arrive at the likelihood of the triple $(d, w, l)_i$

$$P((d,w,l)_i|\zeta) = \sum_{z_i=1}^{K} P(w_i|l_i, z_i, \zeta)P(l_i|d_i, \zeta)P(z_i|d_i, \zeta)P(d_i|\zeta) \quad (6.4)$$

$$= \exp(\delta_{d_i}) \exp(\lambda_{d_i,l_i}) \sum_{z_i=1}^{K} \exp(\omega_{z_i,l_i,w_i}) \exp(\theta_{d_i,z_i}) \quad (6.5)$$

6.3.3 Prior

A uniform prior is used for language-mixture proportions and document probabilities. The prior for the remaining parameters is a product of a prior for logarithmic topic-mixture proportions $\vec{\theta}$ and a prior for logarithmic word-topic associations $\vec{\omega}$.

$$P(\zeta) = P(\vec{\theta}) \cdot P(\vec{\omega}) \quad (6.6)$$

These priors are very similar to those which were defined for PLSA in Section 2.5.3. The prior over logarithmic topic-mixture proportions is a product of N Dirichlet distributions.

$$P(\vec{\theta}) = \prod_{d=1}^{N} \mathrm{Dir}(\vec{\theta}_d|\alpha) \quad (6.7)$$

$$\mathrm{Dir}(\vec{\theta}_d|\alpha) \propto \prod_{k=1}^{K} \exp(\theta_{d,k})^{\alpha/K} \quad (6.8)$$

A product of Dirichlet distributions, one Dirichlet $\mathrm{Dir}(\vec{\omega}_{k,l}|\beta)$ for each topic realization, is used as prior for logarithmic word-topic associations.

$$P(\vec{\omega}) = \prod_{l=1}^{L} \prod_{k=1}^{K} \mathrm{Dir}(\vec{\omega}_{k,l}|\beta) \quad (6.9)$$

$$\mathrm{Dir}(\vec{\omega}_{k,l}|\beta) \propto \prod_{w=1}^{M_l} \exp(\omega_{k,l,w})^{\beta/KM_l} \quad (6.10)$$

The Dirichlets have been transformed to fit to the logarithmic domain of the model parameters. As it was done in case of PLSA, the Dirichlet hyper-parameters are deduced with the help of the principle of equivalent sample-size [19]. As a result, the exponents of the Dirichlets are α/K and β/KM_l. More details about the transformed Dirichlets and the principle of equivalent sample-size can be found in Section 2.5.3.

6.3.4 Learning

Parameter learning is done by applying the Maximum-A-Posteriori (MAP) principle. MAP optimal parameters are those which have a maximal a-posteriori probability.

$$\zeta^* = \underset{\zeta}{\operatorname{argmax}}\, P(\zeta|\vec{X}) \tag{6.11}$$

The number of word occurrences in polylingual document d is denoted by N_d and the number of these from language l is denoted by $N_{d,l}$. MAP-estimates of document probabilities and language-mixture proportions can be derived directly and read as follows.

$$\exp(\delta_d^*) = \frac{N_d}{|\vec{X}|} \tag{6.12}$$

$$\exp(\lambda_{d,l}^*) = \frac{N_{d,l}}{N_d} \tag{6.13}$$

The EM algorithm is used for learning the remaining parameters. This algorithm is an iterative procedure that continuously runs through an E step, in which it determines posteriors for the unobserved topic-index variables, and a M step, in which it re-estimates model parameters. The derivations of this EM algorithm are very similar to those of the EM algorithm for PLSA as explained in more detail in Section 2.6. Thus, the derivations of the EM algorithm for the polylingual topic model are given in the following in a rather condensed manner.

E step

Posteriors $\gamma_{i,k}^{(t+1)} := P(z_i = k|(d,w,l)_i, \zeta^{(t)})$ of the hidden variables z_i with $1 \leq i \leq |\vec{X}|$ and $1 \leq k \leq K$ are derived in the E step of the $(t+1)^{\text{th}}$ EM iteration. The current model parameters $\zeta^{(t)}$, which were determined during the last M step, are used for these computations.

$$\gamma_{i,k}^{(t+1)} \propto P((d,w,l)_i|z_i = k, \zeta^t) P(z_i = k|\zeta^{(t)}) \tag{6.14}$$

$P((d,w,l)_i|z_i = k, \zeta^{(t)})$ is the likelihood of the triple $(d,w,l)_i$ under the assumption that the word w_i was drawn from topic k. $P(z_i = k|\zeta^{(t)})$ is identified with the probability of topic k in EM iteration (t) and is managed by the EM algorithm.

M step

The model parameters are re-estimated by maximizing the expectation of their a-posteriori probability.

$$(\zeta^*)^{(t+1)} := \underset{\zeta^{(t+1)}}{\operatorname{argmax}}\, \mathbb{E}\left[\log P(\vec{X}, \vec{Z}|\zeta^{(t+1)}) + \log P(\zeta^{(t+1)})\right]_{P(\vec{Z}|\zeta^{(t)}, \vec{X})} \tag{6.15}$$

The expectation is defined with respect to the previously computed posteriors $\gamma_{i,k}^{(t+1)}$ of the hidden variables. The previous model parameters $\zeta^{(t)}$ are taken into account through these posteriors. Applying the method of Lagrange multipliers, one finds the following

re-estimates of the model parameters which maximize this expectation.

$$\exp(\theta^*_{d,k})^{(t+1)} \propto \frac{\alpha}{K} + \sum_{\substack{1 \leq i \leq |\vec{X}|, \\ d_i = d}} \gamma_{i,k} \tag{6.16}$$

$$\exp(\omega^*_{w,k,l})^{(t+1)} \propto \frac{\beta}{K M_l} + \sum_{\substack{1 \leq i \leq |\vec{X}|, \\ w_i = w, \\ l_i = l}} \gamma_{i,k} \tag{6.17}$$

The ranges of the word IDs w depend on the language as follows: for each $1 \leq l \leq L$: $1 \leq w \leq M_l$

Commonly, one lets the EM algorithm run continuously through the E and M steps until some break condition stops it. Examples for break conditions are the number of EM iterations already passed or the improvement of the a-posteriori-probability of the model parameters. The parameters of the last M step are then used to instantiate a polylingual topic model.

6.4 Folding-in documents

Folding-in of new documents means to determine their topic-mixture proportions. This is necessary, as, after learning, the polylingual topic model consists of topic-mixture proportions only for the training documents. To assign a probability to a yet unseen document, the topic-mixture proportions of this new document have to be determined. Folding-in a document into a learned polylingual topic model is done as folding-in a document into a learned PLSA model as described in Section 3.3.1. The procedure is to hold the word-topic associations of the learned topic model fixed while running the EM algorithm to determine the topic-mixture proportions of the new document.

6.5 Experiments

The two-dimensional ^{1}H/^{13}C HMBC spectra and bins are shortly called 2D NMR spectra and bins in the rest of this chapter. The effectiveness of the bilingual topic model to predict atom sequences from 2D NMR spectra and to predict 2D NMR bins from atom sequences is investigated in a first experiment. In a second experiment, it is studied how well these predicted atom sequences and 2D NMR bins are suited for information retrieval via look-ups in chemical databases. Before the experimental setup and the results are presented, the data that were used for these experiments are described next.

6.5.1 Data

The 122917 chemical compounds which are listed in the natural-products part of the Beilstein database (version 2006) were used. Compounds that contained other atoms than proton (H), carbon (C), nitrogen (N), and oxygen (O) were omitted to exclude compounds with a very specific chemistry. As these four atom types are by far the most prevalent building blocks of natural compounds, the number of compounds was reduced only to a minor degree by this constraint.

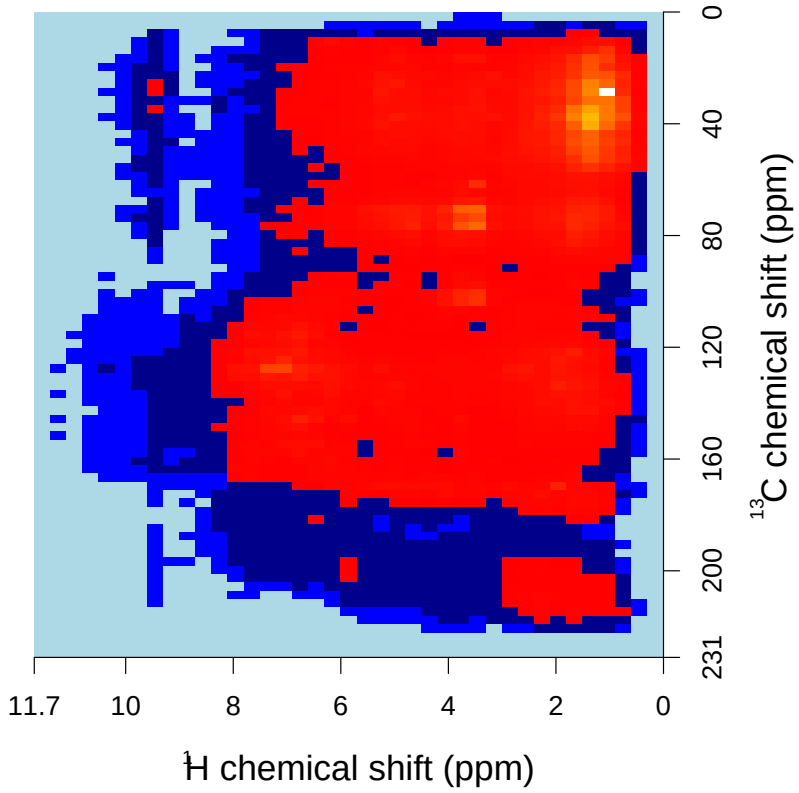


Figure 6.7: Two-dimensional histogram of all simulated ^{1}H/^{13}C HMBC spectra. Each bin is colored according to the overall number of unit peaks that fall into this bin. Light blue 0..9, blue 10..99, dark blue 100..999, red $>$ 1000.

First, atom sequences of length 1, 2, 3 and 4 were determined from the chemical constitution of each chemical compound as described in Section 6.2.1. Each non-proton was chosen once as start-atom and all atom sequences that begin with this start-atom were determined. As a consequence of this approach, each atom sequence was collected twice in forward and backward orientation (order of its atoms). To identify same atom sequences, all derived atom sequences were put into that orientation which led to the lexicographically smaller atom-sequence word. After having determined all atom sequences, the number of occurrences of each atom sequence was corrected by multiplying it by 0.5.

Second, a ^{1}H/^{13}C HMBC spectrum was simulated[6] for each chemical compound with the help of the 2D NMR predictor software of the ACD[7]/Labs software (version 10.0.). Simulated instead of experimentally measured HMBC spectra were used because the latter procedure would have been practically infeasible for the large number of chemical compounds.

The 2D NMR spectra contained peaks of unit intensity for each considered interaction between one proton and one carbon atom. Several equal interactions led to several peaks of unit intensity with the same position. Figure 6.7 shows a summary of all simulated HMBC spectra; large parts of the two-dimensional space of the 2D NMR spectra are

[6]Parameter setting: experiment = HMBC, frequency (MHz) for ^{1}H and ^{13}C = (600.00, 150.87), nucleus = (1H, 13C), Origin ACD/SpecManager (v.10.08), ACD/HNMR Library (v.10.05), ACD/CNMR Library (v.10.05), points count = (512, 512), sweep width (Hz) = (6900.00, 39225.55)

[7]Advanced Chemistry Development

Table 6.1: Sizes of the vocabularies of the two different 2D NMR languages and of the four different atom-sequence languages (AS: atom sequences).

2D-NMR		**AS**	
Bins (^{1}H $\times^{13}$ C)	**size**	**AS length**	**size**
0.05×3	11898	1	112
0.1×6	3367	2	645
		3	2831
		4	11132

populated by simulated peaks. As described in Section 6.2.2, the simulated spectra were discretized into regular 2D NMR bins. Two different 2D NMR languages were used; bins of size 0.05×3 and 0.1×6 whereby the former/latter values indicate the bin size in the ^{1}H/^{13}C dimension. The 2D NMR bins have different lengths in the two dimensions as the overall range of peak positions differ in these two dimensions. The frequency of each deduced 2D NMR word is equal to the number of unit peaks within the corresponding 2D NMR bin.

The determined four different atom-sequence and two different 2D NMR documents per chemical compound were further preprocessed as follows. All words with an absolute document frequency[8] smaller than 5 were removed from the vocabularies. Next, documents with less than 10 different words were omitted. Last, all chemical compounds (and their documents) were removed from the data if some of their six documents were lost during this preprocessing. As a result, a number of 104393 chemical compounds remained in the data. The final sizes of the six different vocabularies are listed in Table 6.1.

6.5.2 Experimental setup

First of all, the training and test compounds were determined. 70% of all chemical compounds, i.e., 73076, were drawn with equal probability. Their documents were used as training documents whereas the documents of the remaining 31317 compounds became test documents. These training and test compounds and documents were held fixed throughout all experiments.

For each chemical compound, eight different versions of bilingual documents were defined. These resulted from all possible combinations of one 2D NMR document (two languages) and one atom-sequence document (four languages) which both were derived from the same chemical compound. As a result, eight different training and test data sets were obtained, one for each language combination. All experiments were repeated eight times, and each time one of these (training and corresponding test) data sets was used, in order to investigate the influence of the different language combinations.

The common design principle of all experiments is visualized in Figure 6.8. First, a bilingual topic model was learned with bilingual training documents which are visualized as green blocks in Figure 6.8. During learning, the information about which 2D NMR and

[8]number of documents they occur in

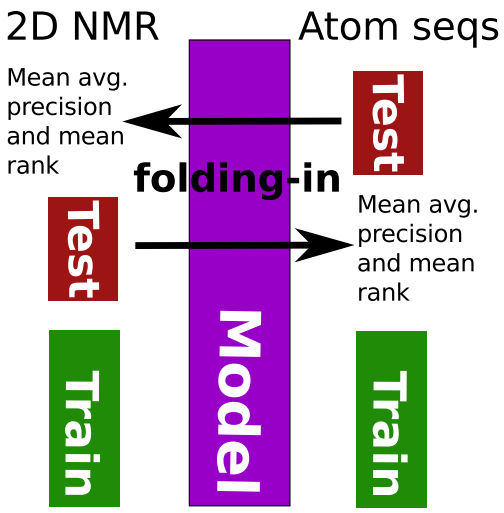


Figure 6.8: Design principle of experiments. Bilingual topic model is learned with training data that consist of pairs of corresponding documents (one atom-sequence and one 2D-NMR document for each chemical compound of the training data). Afterwards, 2D NMR test documents and atom-sequence test documents are folded-in independently of each other. That means, during folding-in information about which 2D NMR test documents corresponds to which atom-sequence test document is not exploited. This information is used afterwards for the computation of mean average precisions (first experiment) and mean ranks (second experiment).

atom-sequence training document describe the same chemical compound was exploited. The number of topics of the bilingual topic model was varied in $\{4, 10, 20, 40, 100, 200, 400\}$. For parameter learning, the EM algorithm was run 50 iterations, it was restarted 3 times, and the hyper-parameters were set to $\alpha = 1$ and $\beta = 1$.

After learning, the test documents were folded-in into the model. In contrast to learning, test documents are folded-in independently of each other. That means, each document, be it a 2D NMR or an atom-sequence document, is folded-in independently of its corresponding document written in the other language. Technically, all 2D NMR and atom-sequence test documents got a unique document ID. But, for the purpose of later use, the information about which 2D NMR test document corresponds to which atom-sequence test document was kept. The EM algorithm with unchanged settings (number of learning iterations, hyper-parameters) was used for folding-in the test documents. After folding-in, the bilingual topic model contained topic-mixture proportions for the 73076 bilingual training documents, for 31317 2D NMR test documents, and for 31317 atom-sequence test documents.

Prediction of 2D NMR bins from atom-sequences and vice versa

In a first experiment, it was investigated how well the bilingual topic model is suited for predicting 2D NMR bins from atom-sequence documents and for predicting atom sequences from the 2D NMR documents. For an atom-sequence test document, 2D NMR bins were predicted as follows. All 2D NMR bins of the whole 2D NMR vocabulary were ranked according to decreasing probability with respect to the predicted distribution

true document	:	a b c a
predicted ranking	:	[a] [c] d [b] e
precisions	:	1 1 0.75
average precision	:	0.92

Figure 6.9: Determination of average precision. Example vocabulary is {a,b,c,d,e}. True hits (framed) in the predicted ranking are those which occur in the true document. The precision of a word w in the ranking is the fraction of true hits which are found in the ranking from the beginning until word w occurs. For example, the precision for word b is $3/4$ as the ranking a c d b until word b contains three true hits. The average precision is the average of all precisions of true hits.

over 2D NMR bins. This predicted distribution was specific for each atom-sequence test document. In more detail, the bilingual topic model contained logarithmic topic-mixture proportions $\vec{\theta}_d$ for each atom-sequence test document d. The predicted distribution over 2D NMR bins (language ID is $l = 2$) for this atom-sequence test document is $P(w|d, l = 2) = \sum_{k=1}^{K} \exp(\theta_{d,k}) \cdot \exp(\omega_{k,l=2,w})$ with $1 \leq w \leq M_2$ and M_2 is the size of the 2D NMR vocabulary. The predicted ranking of 2D NMR bins was then compared to the true 2D NMR bins which were those of the 2D NMR test document that did correspond to the atom-sequence document. The predicted ranking was assessed by the average precision of all true 2D NMR bins in this ranking. The determination of the average precision is illustrated in Figure 6.9. This procedure was applied to all atom-sequence test documents and the mean over all average precisions was computed. The mean average precision measures how well, on average, works the prediction of 2D NMR bins from atom-sequence documents. A similar procedure was applied to all 2D NMR test documents and the mean average precision of the prediction of atom sequences from 2D NMR documents was determined.

Cross-language retrieval of chemical compounds

The second experiment gives clues about how well the predicted 2D NMR bins and atom sequences are suited for cross-language retrieval for chemical compounds. Two potential application scenarios are

1. To search in chemical databases, which store only the chemical constitution but no NMR data, for chemical compounds that are similar to a newly investigated compound with yet unknown constitution but known 2D NMR spectrum.

2. To search in NMR databases, which store NMR data but no information about the chemical constitution, for NMR spectra of chemical compounds which are identical or similar to a query compound.

The learned bilingual topic models which have been extended by the test documents in the first experiment were used during the second experiment. Randomly chosen 1000 test compounds have been designated as look-up compounds which were used in this experiment. All other test documents were neglected in order to reduce the computational costs.

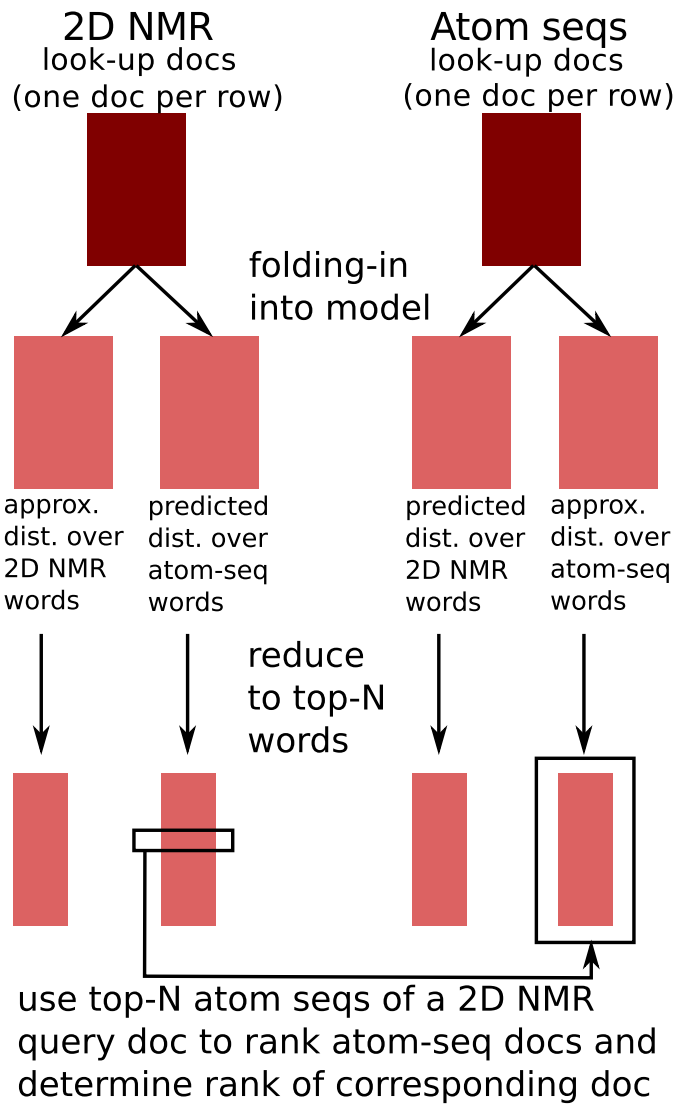


Figure 6.10: Design of experiment for cross-language compound retrieval. The designated 1000 look-up documents are visualized as colored blocks (matrices; one document per row). The visualized task of ranking atom-sequence documents is an example of cross-language compound retrieval with a 2D NMR query document and the targets are the atom-sequence documents.

The experimental design of the second experiment is visualized in Figure 6.10. The 1000 2D-NMR look-up documents and their corresponding 1000 atom-sequence look-up documents have been folded-in into the model independently of each other. For each of these 2000 documents, the bilingual topic model (i) approximates their word distribution (same language), and (ii) predicts a distribution over words of the other language. For example, for an atom-sequence document d with logarithmic topic-mixture proportions $\vec{\theta}_d$ the approximated distribution over atom-sequence words (language ID $l = 1$) is $P(w|d, l = 1) = \sum_{k=1}^{k} \exp(\theta_{d,k}) \cdot \exp(\omega_{k,l=1,w})$ with $1 \leq w \leq M_1$ and M_1 is the size of the atom-sequence vocabulary. The predicted distribution over 2D NMR bins was described together with the first experiment.

The approximated distributions over words of the own language and the predicted distribution over words of the other language were used to derive fingerprint representations for each of the 2000 look-up document. In more detail, the top-5, top-35, top-65, and top-100 words which are most likely under the approximated and predicted word distributions are used as fingerprints. In addition, the topic-mixture proportions of the 2000 look-up documents were used as a second kind of fingerprint representation.

These fingerprints were then used for cross-language compound retrieval. That means, for a query 2D NMR document the goal was to find the true corresponding atom-sequence document among all 1000 look-up atom-sequence documents. This was done by comparing the predicted atom-sequence fingerprint of the query 2D NMR document to all atom-sequence fingerprints of the 1000 atom-sequence look-up documents. Top-N fingerprints, which are sets of words, were compared to each other by measuring their similarity with the help of the Jaccard coefficient. Formally, the Jaccard coefficient for two sets A and B is defined as

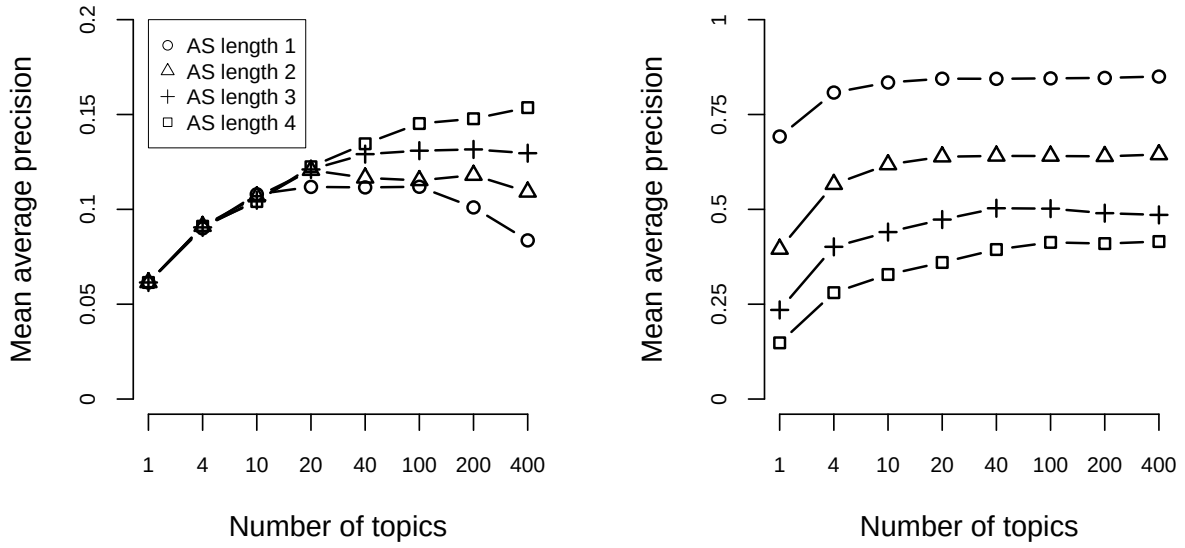
$$J(A, B) := \frac{|A \cap B|}{|A \cup B|} \quad .$$

This coefficient takes values between 0 and 1 and measures how well two top-N fingerprints (set of words) agree. All look-up documents of the target language were ranked according to decreasing Jaccard coefficients. Topic-mixture proportions were compared to each other by computing their KL divergence. In this case, the look-up documents of the target language were ranked according to increasing KL divergences. Each of the 1000 look-up 2D NMR documents was once used as a query document and the rank of the true corresponding atom-sequence documents was determined. The mean of these 1000 ranks was reported. A similar procedure was applied to all 1000 atom-sequence look-up documents, which were used as query documents, for the search of the corresponding 2D NMR document.

6.5.3 Results

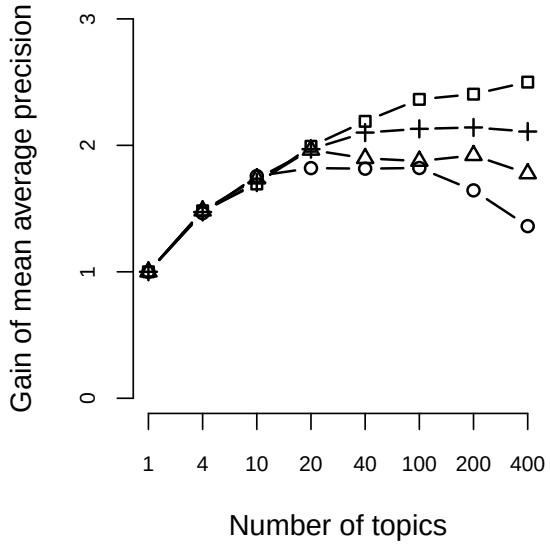
Prediction of 2D NMR bins from atom-sequences and vice versa

The results of the first experiment are shown in Figure 6.11/6.12 for the 2D-NMR language of bins of size $0.05 \times 3/0.1 \times 6$, respectively. Figure 6.11 shows results for the four combinations of the 2D NMR language (bins of size 0.05×3) with each of the four atom-sequence languages. Figure 6.11(a) presents the reached mean average precisions (MAP) for the prediction of 2D NMR words from the atom-sequence test documents. A

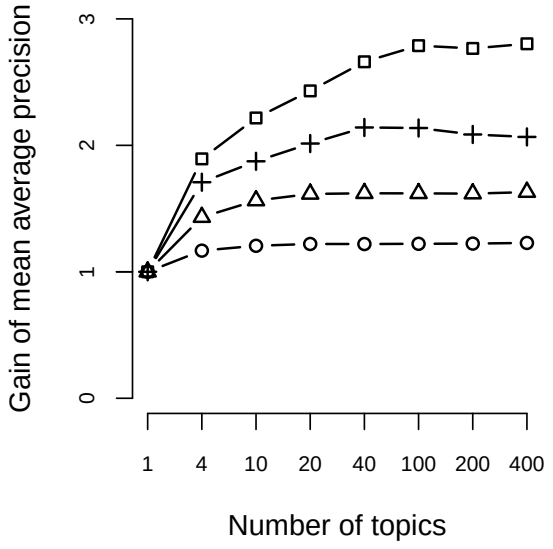


(a) NMR from AS

(b) AS from NMR

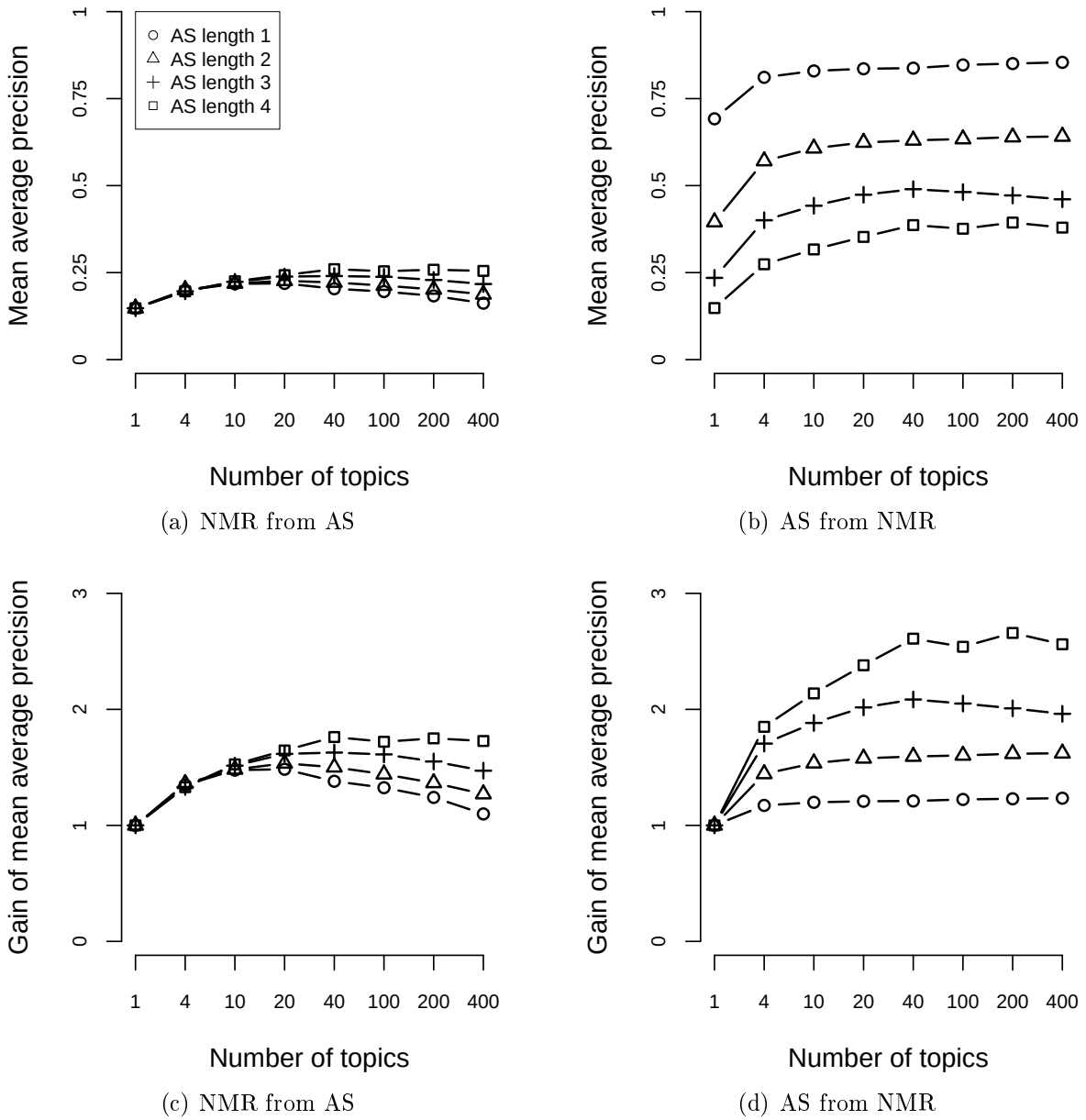


(c) NMR from AS



(d) AS from NMR

Figure 6.11: Mean average precisions (MAP) for the prediction of 2D NMR words (2D NMR bins of size 0.05×3) from atom sequences (AS) (**left**) and vice versa (**right**). A number of one topic stand for the bilingual unigram model (one topic per language). The gain of MAP was computed with respect to the reached MAP for the unigram model (**bottom**).



(a) NMR from AS

(b) AS from NMR

(c) NMR from AS

(d) AS from NMR

Figure 6.12: Mean average precisions (MAP) for the prediction of 2D NMR words (2D NMR bins of size 0.1×6) from atom sequences (AS) (**left**) and vice versa (**right**). A number of one topic stands for the bilingual unigram model (one topic per language). The gain of MAP was computed with respect to the reached MAP for the unigram model (**bottom**).

simple bilingual unigram model was learned as a reference model (one topic). Using this reference model leads to the worst MAP of about 0.06 for all language combinations. The curves for the languages of atom sequences of length 1, 2, and 3 have a concave shape. In contrast thereto, the MAPs for atom sequences of length 4 steadily increase. The optimal MAPs per curve are 0.112 (20 topics), 0.121 (20 topics), 0.132 (200 topics), and 0.154 (400 topics) for atom sequences of length 1, 2, 3, and 4, respectively. These results show that the optimal MAPs increase with the length of the atom sequences. An explanation could be that longer atom sequences capture the atom topology of chemical constitutions better. As such, they contain more information which seems to be useful for the prediction of 2D NMR words. A second trend is that the longer the atom sequences are, the more topics are necessary to reach optimal MAPs. A reason could be that the size of the vocabulary of atom sequences increases with their length. Patterns of co-occurring atom sequences as captured by the bilingual topics might become more complex with an increasing vocabulary. As models with a larger number of topics have a larger statistical expressiveness, these might be better suited for capturing the more complex patterns.

The same trends of the measured MAPs are found for the prediction of 2D NMR bins of size 0.1×6 in Figure 6.12(a). The most obvious difference is that these MAPs are generally larger compared to the MAPs for the prediction of the smaller 2D NMR bins. The optimal MAPs per curve are 0.219 (20 topics), 0.226 (20 topics), 0.24 (40 topics), and 0.258 (200 topics) for atom sequences of length 1, 2, 3, and 4, respectively. A reason for the larger MAPs might be the smaller size of the vocabulary of 2D NMR bins of size 0.1×6 compared to the language of the smaller 2D NMR bins. As presented in Table 6.1, the vocabulary size of the latter language is 11898 whereas it is 3367 for 2D NMR bins of size 0.1×6. Obviously, correctly ranking 2D NMR words is easier, on average, when the number of different 2D NMR words is smaller.

Next, we focus on the prediction of atom sequences from 2D NMR test documents. Figure 6.11(b) shows MAPs for the language combinations of 2D NMR bins of size 0.05×3 and atom sequences of length 1 to 4. The MAPs for the simple unigram model vary according to the length of the atom sequences. In addition, the shorter the atom sequences the larger the reached MAPs. A reason for this observation seems to be the different sizes of the atom-sequence languages; the shorter the atom sequences the smaller the vocabulary size and the easier it is, on average, to correctly predict atom sequences. Studying the obtained MAPs in more detail, the optimal MAPs are: 0.85 (400 topics), 0.644 (400 topics), 0.503 (40 topics), and 0.415 (400 topics) for atom sequences of length 1, 2, 3, and 4, respectively. Is seems that the bilingual topic models with the most topics are often best suited for the prediction of atom sequences from 2D NMR documents. For comparing these MAPs among each other, the gain of MAP in relation to the reference models are presented in Figure 6.11(d). It becomes obvious that, in relation to the reference models, the longer the atom sequences are, the more their prediction benefits from bilingual topic modeling. A reason might be that the number of different atom sequences increases with their length and so become the patterns of their co-occurrences more complex. These more complex patterns then might be better modeled by bilingual topic models that consists of a larger number of topics. Another interesting finding is the prediction of atom sequences from 2D NMR documents seems to work better (higher MAPs) than the reverse prediction. To compare both predictions, we focus on the optimal MAP of 0.415 for atom sequences of length 4 in Figure 6.11(b). This MAP

can be compared to the optimal MAP of 0.154 for the prediction of 2D NMR words from atom sequences of length 4 in Figure 6.11(a). Both optimal MAPs can be compared to each other as the vocabularies of 2D NMR bins of size 0.05×3 and of atom sequences of length 4 are similar in size (about 11000) as listed in Table 6.1. A reason for the higher accuracy of the prediction of atom sequences could be that NMR signals are rich in information about the chemical constitution of a chemical compound. On the other hand, atom sequences capture only fragments of the chemical constitution. Hence, the prediction of 2D NMR words from atom sequences is more challenging.

The results for the prediction of atom sequences from the larger 2D NMR bins of size 0.1×6 are presented in Figure 6.12(b). These MAPs show the same trends as the previous results, and so confirm the previous findings, which were obtained for the smaller 2D NMR bins. We find the following optimal MAPs: 0.854 (400 topics), 0.64 (400 topics), 0.49 (40 topics), and 0.394 (200 topics) for atom sequences of length 1, 2, 3, and 4, respectively. Interestingly, for each length of atom sequences we find a similar best MAP for the prediction from 2D NMR words of size 0.05×3 and 0.1×6. Hence, the prediction of atom sequences from 2D NMR bins seems not to benefit from the finer resolution of the smaller 2D NMR bins. This was unexpected as a finer 2D NMR resolution would encode the positions of the 2D NMR signals more precisely and, hence, was expected to result in better predictions of atom sequences. As seen previously, the prediction of atom sequences seems to work better than the prediction of 2D NMR bins. The optimal MAP of 0.49 for the prediction of atom sequences of length 3 (vocabulary size 2831) is larger than the optimal MAP of 0.24 for the prediction of 2D NMR bins of size 0.1×6 (vocabulary size 3367) from atom sequence of length 3. An even stronger indication is that the optimal MAP of 0.394 for the prediction of atom sequences of length 4 is larger than the optimal MAP of 0.24 for the prediction of 2D NMR bins from atom sequence of length 3. This is notably as the vocabulary size of atom sequences of length 4 (11132) is by far larger than the vocabulary size of 2D NMR bins of size 0.1×6 (3367). Hence, although it is harder to predict atom sequences of length 4 due to their larger vocabulary size, this prediction leads to better MAPs than the prediction of 2D NMR bins of size 0.1×6.

In summary, the prediction of atom sequences from 2D NMR bins seems to work better than the reverse prediction of 2D NMR bins from atom sequence documents. The prediction of 2D NMR bins benefits from describing the chemical constitution by atom sequences of longer length which capture the topology of the chemical constitution better. On the other hand, the prediction of atom sequences seem not to benefit from a finer resolution of the 2D NMR language. We find that the reached MAPs for the prediction of atom sequences from 2D NMR bins are similar for both 2D NMR languages of bins of size 0.05×3 and 0.1×6.

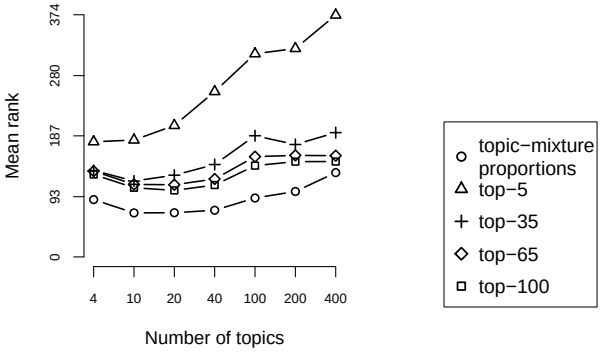
Cross-language retrieval of chemical compounds

The results of the second experiment are presented in Figure 6.13/6.14 for the 2D NMR language of bins of size $0.05 \times 3 / 0.1 \times 6$. Both figures are organized as follows. The results of the left column are mean ranks for the search with atom-sequence query documents among 2D NMR documents. The results which are shown in the right column are those for the reverse cross-language retrieval with 2D NMR query documents. The four different rows correspond to the four combinations of the 2D NMR language and the

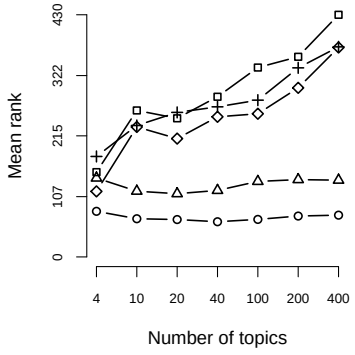
language of atom sequences of length 1 to 4.

When studying the search with atom-sequence query documents among 2D NMR documents (left column of Figure 6.13), we find the following general pattern. Representing documents by the top-5 2D NMR words by far leads to worst mean ranks. The more top N 2D-NMR words are used as document fingerprints, the better (lower) the mean rank becomes. Second, using topic-mixture proportions as document fingerprints, we obtain the best mean ranks. These statements are true throughout the whole left column of Figure 6.13. A reason for the good performance of the topic-mixture proportions could be that they represent documents by topic weights instead of predicted words. Topic weights are a more general document property which abstracts from single word occurrences. As such, topic weights might be better suited for the detection of similarities among documents. In addition, the predicted top-N words might be contaminated by too many falsely predicted words which hinder cross-language document retrieval. Further on, the results indicate that a bilingual topic model with a medium number of topics from 20 to 40 gives best mean ranks. For example, mean ranks for top-35, top-65, top-100 and topic-mixture proportions are minimal for 20 topics in Figures 6.13(c) and 6.13(e) and for 40 topics in Figure 6.13(g). These mean ranks seem to indicate that, if longer atom sequences are used, a bilingual topic model with a slightly larger number of topics leads to best mean ranks. The reason might be that patterns of co-occurring atom sequence become more complex if the size of the atom-sequence vocabulary increases. Consequently, a larger number of topics is necessary for capturing these patterns. Another observation is that mean ranks become better if longer atom sequences are used. The reason could be that longer atom sequences capture the topology of the chemical constitution better and so allow to better predict 2D NMR bins. Document fingerprints that consist of better predicted 2D NMR bins then might lead to better retrieval of 2D NMR documents. Interestingly, cross-language compound retrieval via topic-mixture proportions benefits from longer atom sequences, too. This becomes obvious when inspecting the best reached mean ranks for topic-mixture proportions as listed in Table 6.2. A reason could be that the patterns of co-occurring atom sequences of longer length are more meaningful as these better encode the chemical constitution. With more meaningful bilingual topics, which span the topic space, coordinates of topic-mixture proportions, which lie in the topic space, could reflect similarities among chemical compounds better. Last, the overall best mean rank throughout the left column of Figure 6.13, is 30 (Figure 6.13(g)). This best mean rank is obtained when representing documents by their topic-mixture proportions and when using atom sequences of length 4 for the description of the chemical constitution.

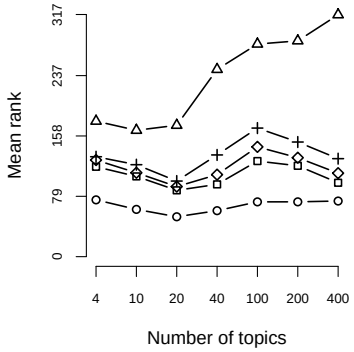
When studying the right column of Figure 6.13, we again find, in accordance with the left column, that 20 to 40 topics often lead to best mean ranks. Further on, topic-mixture proportions give best overall mean ranks. All best mean ranks for topic-mixture proportions and the different combinations of the 2D NMR and atom-sequence languages are listed in Table 6.2. Focusing on the left half (2D NMR bins of size 0.05×3, column AS) of this table, we find the best mean rank of 28 for the search among atom-sequence documents with 2D NMR query documents. This mean rank of 28 is reached when one uses atom sequences of length 4 to encode the chemical constitution of chemical compounds. Studying the left half of Table 6.2 closer, the following trend becomes obvious. The retrieval among the atom-sequence documents for a query 2D NMR document reaches better (lower) mean ranks (column AS) than the reverse retrieval task (column



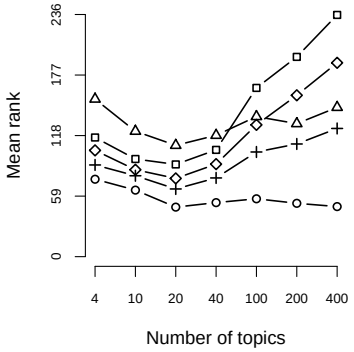
(a) query: AS1, target NMR



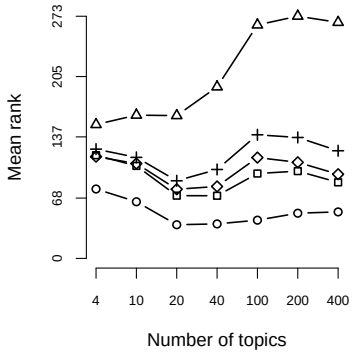
(b) query: NMR, target AS1



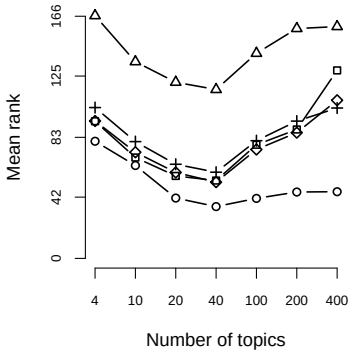
(c) query: AS2, target NMR



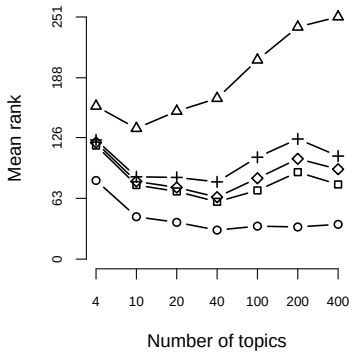
(d) query: NMR, target AS2



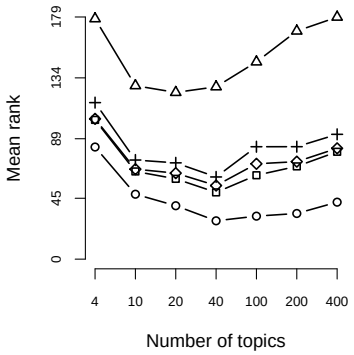
(e) query: AS3, target NMR



(f) query: NMR, target AS3



(g) query: AS4, target NMR



(h) query: NMR, target AS4

Figure 6.13: Mean ranks for the search with atom-sequence query documents among 2D NMR documents (left column), and for the search with 2D NMR query documents among atom-sequence documents (right column). 2D NMR language are bins of size 0.05×3. AS1, ..., AS4 denote atom sequences of length 1 up to 4.

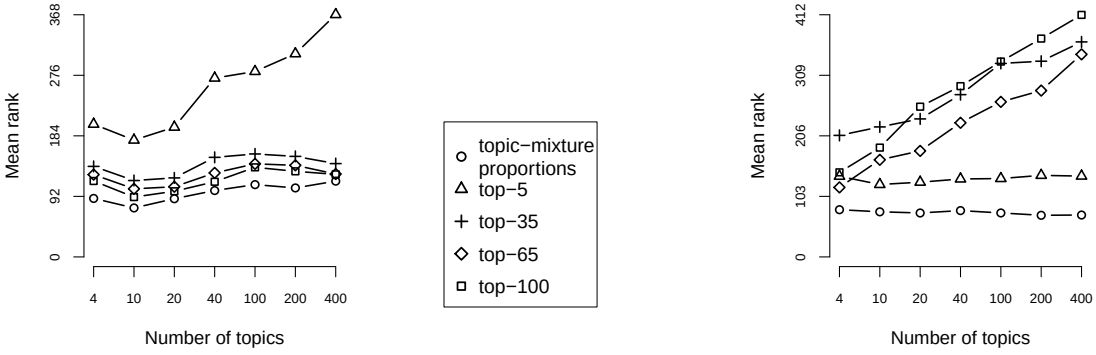
Table 6.2: Lowest (best) mean ranks for cross-language compounds retrieval when topic-mixture proportions are used as document fingerprints. Shown are results for different combinations of a 2D NMR language and an atom-sequence (AS) language. Target language is the language (NMR, AS) of the target documents.

	2D-NMR bins of size 0.05×3		**2D-NMR bins of size 0.1×6**	
target language →	**NMR**	**AS**	**NMR**	**AS**
AS length 1	68	62	74	72
AS length 2	52	48	58	51
AS length 3	38	36	45	42
AS length 4	30	28	33	33

NMR). The best mean ranks for the former task, which are 62, 48, 36, 28, are lower than the corresponding (same atom-sequence language) mean ranks for retrieving 2D NMR documents, which are 68, 52, 38, 30, respectively. This observation is inline with the results of the first experiment where we found that the prediction of atom sequences from 2D NMR words works better than the reverse prediction.

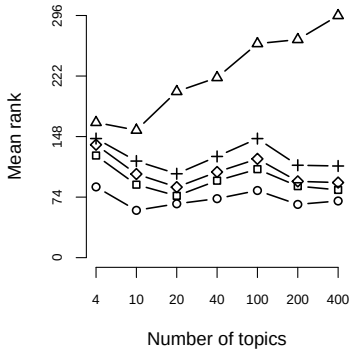
These experiments were repeated using the 2D-NMR language of larger bins of size 0.1×6. The results are presented in Figure 6.14. These are very similar to the previous results and, so, approve the previous findings. Still are topic-mixture proportions best suited for cross-language compound retrieval in the context of modeling chemical compounds. The corresponding best mean ranks, which are presented in Table 6.2 (right half), are often reached when a topic model with a number of 20 to 40 topics is used. The best mean ranks listed in the right half of Table 6.2 are 33/33 for retrieving 2D NMR/atom-sequence documents. These are obtained for atom sequences of length 4. When comparing the left with the right half of Table 6.2, one finds that all mean ranks which are listed at the left half are lower than the corresponding mean ranks listed at the right half of Table 6.2. This indicates that cross-language compound retrieval with the help of topic-mixture proportions benefits from a finer resolution of the 2D NMR spectra.

In summary, topic-mixture proportions are clearly superior over predicted top-N words for cross-language compound retrieval. Second, the retrieval of atom-sequence documents for a query 2D NMR document is accomplished with slightly better performance than the reverse retrieval. Third, the best mean ranks are 30 and 28 for the retrieval of 2D NMR and atom-sequence documents, respectively. Both results are obtained when 2D NMR documents are encoded with a finer resolution by 2D NMR bins of size 0.05×3 and when atom-sequence documents are encoded by atom sequences of length 4. Further on, these best mean ranks are obtained with a bilingual topic model with a moderate number of 40 topics.

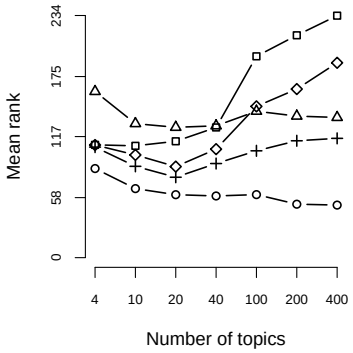


(a) query: AS1, target NMR

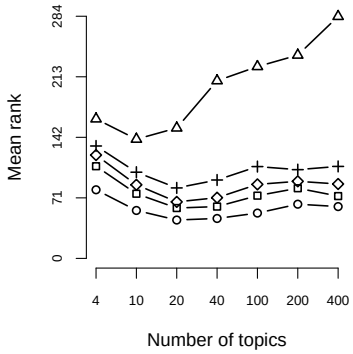
(b) query: NMR, target AS1



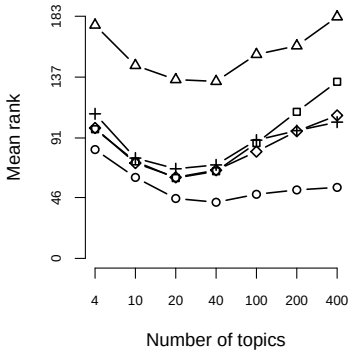
(c) query: AS2, target NMR



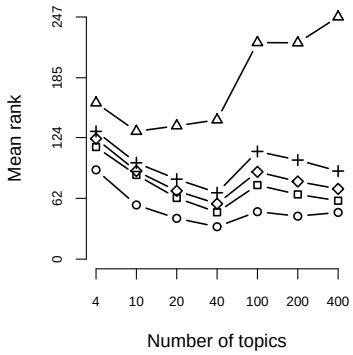
(d) query: NMR, target AS2



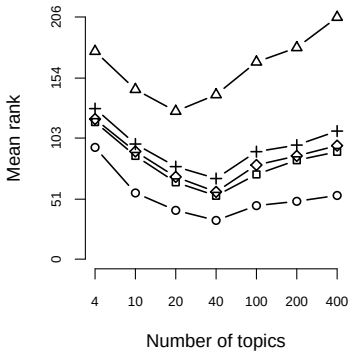
(e) query: AS3, target NMR



(f) query: NMR, target AS3



(g) query: AS4, target NMR



(h) query: NMR, target AS4

Figure 6.14: Mean ranks for the search with atom-sequence query documents among 2D NMR documents (left column), and for the search with 2D NMR query documents among atom-sequence documents (right column). 2D NMR language are bins of size 0.1×6. AS1, ..., AS4 denote atom sequences of length 1 up to 4.

6.6 Conclusions and future directions

Bilingual topic modeling of chemical compound is an attempt to jointly model two perspectives on chemical compounds. These perspectives are (i) spectra of NMR experiments, and (ii) the chemical constitution of the chemical compounds. To apply the bilingual topics model, the 2D NMR spectra are discretized into bins which become words of the 2D NMR language. A 2D NMR word is observed whenever a 2D NMR signal falls into a 2D NMR bin. The chemical constitution is broken down to linear sequences of connected atoms which are called atom sequences. These atom sequences become the words of the atom-sequence language. A bilingual topic model might be applied to the 2D NMR and an atom-sequence documents which have been derived from a training set of chemical compounds. Learning this bilingual topic model means to learn patterns of 2D NMR bins and atom sequences which often occur in the two different documents which have been derived form the same chemical compounds. For a test compound, a learned bilingual topic model makes it possible to predict 2D NMR bins from its chemical constitution and to predict atom sequences from its 2D NMR spectrum.

In a first experiment, these predictions were assessed by the average precision of the true hits among the predicted 2D NMR bins or atom sequences. It turns out that the longest tested atom sequences of length 4 give best average precisions for the prediction of 2D NMR bins. A reason might be that longer atom sequences better capture the topology of the chemical constitution. As such, they are richer in information about the electromagnetic interactions among atoms of the chemical compounds. As a result, they might be more useful for the prediction of the 2D NMR bins. In contrast, the average precisions for the prediction of atom sequences from 2D NMR documents are not affected by different resolutions of the discretization of 2D NMR spectra. Another general observation is that the prediction of atom sequences from 2D NMR bins, on average, works better than the prediction of 2D NMR bins from atom-sequence documents.

In a second experiment, the predicted top-N 2D NMR bins, the predicted top-N atom sequences and the topic-mixture proportions were used for cross-language compound retrieval. Topic-mixture proportions of a moderate number of 40 topics lead to best mean ranks for the search among 2D NMR documents with atom-sequence query documents as well as for the reverse retrieval. A reason might be that topic-mixture proportions represent documents by more abstract properties than the predicted top-N words do. In addition, predicted top-N words might include falsely predicted words which might reduce the effectiveness of the top-N words for cross-language compound retrieval. The optimal mean rank for the retrieval of 2D NMR documents with atom-sequence query documents is 30. For the reverse retrieval task, the optimal mean rank is 28. These optimal mean ranks indicate that the approach of bilingual topic modeling is promising for supporting chemical research.

An open future route is to closer investigate the rankings of the cross-language compound retrieval experiment. A mean rank of 30 means that, on average, the true compound was ranked on position 30. The other compounds which are ranked before this true hit could have a totally different chemical constitution. As such, they are correctly classified as false hits. On the other side, the top-ranked compounds might have a very similar chemical constitution. For example, the chemical constitutions might be identical except for a substituted chemical functional group. In this case, top-ranked compounds

that are very similar to the true hit might not be judged as false hits.

One future direction is to describe the chemical constitution of chemical compounds by other structure languages. Atom sequences, as used in this work, describe linearly connected sequences of atoms. Other concepts of structural fragments might be suited better for capturing groups of connected atoms. An example are subsets of connected atoms as shown in Figure 6.2 (right). To this end, concepts from other disciplines like cliques of tightly connected atoms from graph theory could be helpful. The experiments which are proposed in this chapter might be used in order to assess new proposed languages for the description of the chemical constitution.

Another future direction is to enhance the modeling of the 2D NMR spectra. In this work, 2D NMR spectra were discretized such that only those NMR peaks, which lie exactly in a bin, belong to this bin. Peaks, which lie close to a border of a bin, might be wrongly separated and put into different bins. Instead of a strict discretization, one could use a soft discretization where a two-dimensional Gauss distribution is located at the center of each bin. A sum over all NMR peak intensities which have been weighted by this Gauss distribution might then be assigned to each bin. Another approach is to extend the multilingual topic model such that it models 2D NMR spectra as continuous data. In this case, NMR words could be two-dimensional Gauss distributions over the plane of the 2D NMR spectra. Then, each NMR topic could be modeled as a specific mixture of these distributions. Another advantage of modeling the NMR spectra as continuous data might be the reduced number of model parameters.

A third future direction could be to extend the polylingual topic model such that it models additional kinds of data. Beside modeling 2D NMR spectra and the chemical constitution of chemical compounds, data from different NMR experiments, mass spectrometry, infrared spectroscopy might be modeled as well. Learning the polylingual topic model with more kinds of data could be beneficial. For example, exploiting additional sources of information could help to turn an unspecific pattern of 2D NMR signals and structural fragments into several more specific patterns. One challenge of this future direction is to acquire the additional data as this could be expensive in terms of lab resources (materials, instruments), manpower, and financial aspects.

Bibliography

[1] Harcourt Brown. History and the Learned Journal. *Journal of the History of Ideas*, 33(3):365–378, 1972. ISSN 00225037. doi: 10.2307/2709041.

[2] Bo-Christer Björk, Annikki Roos, and Mari Lauri. Global annual volume of peer reviewed scholarly articles and the share available via different Open Access options. In *The International Conference on Electronic Publishing (ELPUB 2008) - Open Scholarship: Authority, Community and Sustainability in the Age of Web 2.0, June 25-27 2008*, June 2008.

[3] Arif E. Jinha. Article 50 million: an estimate of the number of scholarly articles in existence. *Learned Publishing*, 23(3):258–263, July 2010. ISSN 0953-1513. doi: 10.1087/20100308.

[4] David M. Blei. Introduction to probabilistic topic models. Available at his website `http://www.cs.princeton.edu/~blei/papers/Blei2011.pdf`, (last accessed May 11, 2012), 2011.

[5] Andre Gohr, Alexander Hinneburg, Rene Schult, and Myra Spiliopoulou. Topic evolution in a stream of documents. In *SIAM Data Mining Conf. (SDM'09)*, pages 378–385, Reno, CA, Apr.-May 2009.

[6] André Gohr, Myra Spiliopoulou, and Alexander Hinneburg. Visually summarizing the evolution of documents under a social tag. In Ana L. N. Fred and Joaquim Filipe, editors, *Proceedings of International Conference on Knowledge Discovery and Information Retrieval, KDIR*, pages 85–94. SciTePress, 2010. ISBN 978-989-8425-28-7.

[7] André Gohr, Myra Spiliopoulou, and Alexander Hinneburg. Visually summarizing semantic evolution in document streams with Topic Table. In A. Fred et al., editor, *Communications in Computer and Information Science*, volume 223, pages 136–150. Springer-Verlag Berlin Heidelberg, 2012.

[8] André Gohr, Myra Spiliopoulou, and Alexander Hinneburg. Visually summarizing the evolution of documents under a social tag. In Martin Atzmüller, Dominik Benz, Andreas Hotho, and Gerd Stumme, editors, *Proceedings of LWA2010 - Workshop-Woche: Lernen, Wissen & Adaptivität*, 2010.

[9] Dawn E. Holmes. Toward a generalized Bayesian network. In Dawn E. Holmes and Lakhmi C. Jain, editors, *Innovations in Bayesian Networks: Theory and Applications*, volume 156 of *Studies in Computational Intelligence*, pages 281–288. Springer, 2008. ISBN 978-3-540-85065-6.

[10] André Gohr, Alexander Hinneburg, Myra Spiliopoulou, and Ricardo Usbeck. On the distinctiveness of tags in collaborative tagging systems. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, WIMS, pages 62:1–62:5, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0148-0.

[11] André Gohr, Andrea Porzel, Ludger Wessjohann, and Alexander Hinneburg. Bilingual topic modeling of chemical compounds, 2010. Reviewed and accepted for topic model workshop, ICML 2010, Israel.

[12] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using EM, 2000. URL `http://www.springerlink.com/index/P4324Q3673265225.pdf`.

[13] George Tsatsaronis and Vicky Panagiotopoulou. A generalized vector space model for text retrieval based on semantic relatedness. In *EACL '09: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 70–78, Morristown, NJ, USA, 2009. Association for Computational Linguistics.

[14] Thomas Hofmann. Probabilistic latent semantic indexing. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57, New York, NY, USA, 1999. ACM. ISBN 1-58113-096-1. doi: http://doi.acm.org/10.1145/312624.312649.

[15] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003. ISSN 1533-7928.

[16] J.K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945, 2000.

[17] Christopher Cannings D. J. Balding, Martin J. Bishop, editor. *Handbook of statistical genetics*, volume 1. John Wiley & Sons, Ltd., 3 edition, 2007.

[18] Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Mach. Learn.*, 42(1-2):177–196, 2001. ISSN 0885-6125. doi: http://dx.doi.org/10.1023/A:1007617005950.

[19] Dan Geiger and David Heckerman. Parameter priors for directed acyclic graphical models and the characterization of several probability distributions. *Annals of Statistics*, 30:2002, 1999.

[20] J. T. Chien and M. S. Wu. Adaptive Bayesian Latent Semantic Analysis. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(1):198–207, 2008. doi: 10.1109/TASL.2007.909452.

[21] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1): 1–38, 1977. Series B.

[22] B. G. Lindsay. *Mixture Models: Theory, Geometry and Applications*. NSF-CBMS Regional Conference Series in Probability and Statistics, Alexandria, Virginia: Institute of Mathematical Statistics and the American Statistical Association, 1988.

[23] Geoffrey McLachlan and David Peel. *Finite Mixture Models*. Wiley Series in Probability and Statistics. Wiley-Interscience, October 2000. ISBN 0471006262.

[24] Richard A. Redner and Homer F. Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26(2):195–239, 1984. doi: 10.2307/2030064.

[25] Gregor Heinrich. Parameter estimation for text analysis. Technical report, Fraunhofer IGD, Darmstadt, Germany, Sep. 2009. version 15.

[26] Jordan Boyd-Graber, Jonathan Chang, Sean Gerrish, Chong Wang, and David Blei. Reading tea leaves: How humans interpret topic models. In *Neural Information Processing Systems (NIPS)*, 2009.

[27] David M. Blei, Thomas L. Griffiths, Michael I. Jordan, and Joshua B. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, page 2003. MIT Press, 2004.

[28] Zoubin Ghahramani and Matthew J. Beal. Variational inference for Bayesian mixtures of factor analysers. In *In Advances in Neural Information Processing Systems 12*, pages 449–455. MIT Press, 2000.

[29] Hagai Attias. A variational Bayesian framework for graphical models. In *In Advances in Neural Information Processing Systems 12*, pages 209–215. MIT Press, 2000.

[30] David J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, New York, NY, USA, 2002. ISBN 0521642981.

[31] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Publishing Company, Incorporated, 2008. ISBN 0387763694, 9780387763699.

[32] Arthur Asuncion, Max Welling, Padhraic Smyth, and Yee Whye Teh. On smoothing and inference for topic models. In *In Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, 2009.

[33] Mark Girolami and Ata Kabán. On an equivalence between PLSI and LDA. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 433–434, New York, NY, USA, 2003. ACM. ISBN 1-58113-646-3.

[34] Wray Buntine and Aleks Jakulin. Discrete component analysis. In *Subspace, Latent Structure and Feature Selection Techniques*. Springer-Verlag, 2006.

[35] Tom Griffiths. Gibbs sampling in the generative model of Latent Dirichlet Allocation. Technical report, Stanford University, 2002.

[36] Yee Whye Teh, David Newman, and Max Welling. A collapsed variational Bayesian inference algorithm for Latent Dirichlet Allocation. In *In NIPS 19*, pages 1353–1360, 2007.

[37] Ian Porteous, David Newman, Alexander Ihler, Arthur Asuncion, Padhraic Smyth, and Max Welling. Fast collapsed Gibbs sampling for Latent Dirichlet Allocation. In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 569–577, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-193-4.

[38] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235, April 2004.

[39] Qiaozhu Mei and ChengXiang Zhai. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *SIGKDD*, pages 198–207, New York, NY, USA, 2005. ACM. ISBN 1-59593-135-X. doi: http://doi.acm.org/10.1145/1081870.1081895.

[40] D. Blei and J. Lafferty. Dynamic topic models. In *ICML*, 2006.

[41] Xuerui Wang and Andrew McCallum. Topics over time: a non-Markov continuous-time model of topical trends. In *SIGKDD*, pages 424–433. ACM, 2006. doi: http://doi.acm.org/10.1145/1150402.1150450.

[42] Satoshi Morinaga and Kenji Yamanishi. Tracking dynamics of topic trends using a finite mixture model. In *SIGKDD*, pages 811–816. ACM, 2004. doi: http://doi.acm.org/10.1145/1014052.1016919.

[43] Charu C. Aggarwal and Philip S. Yu. A framework for clustering massive text and categorical data streams. In *SDM*, 2006.

[44] Myra Spiliopoulou, Irene Ntoutsi, Yannis Theodoridis, and Rene Schult. Monic: modeling and monitoring cluster transitions. In *SIGKDD*, pages 706–711. ACM, 2006. doi: http://doi.acm.org/10.1145/1150402.1150491.

[45] Rene Schult and Myra Spiliopoulou. Expanding the taxonomies of bibliographic archives with persistent long-term themes. In *SAC*, pages 627–634, 2006.

[46] Rene Schult and Myra Spiliopoulou. Discovering emerging topics in unlabelled text collections. In *ADBIS*, pages 353–366, 2006.

[47] Rene Schult. Comparing clustering algorithms and their influence on the evolution of labeled clusters. In *DEXA*, pages 650–659, 2007.

[48] Xiaodan Song, Ching-Yung Lin, Belle L. Tseng, and Ming-Ting Sun. Modeling and predicting personal information dissemination behavior. In *SIGKDD*, pages 479–488. ACM, 2005. doi: http://doi.acm.org/10.1145/1081870.1081925.

[49] Loulwah AlSumait, Daniel Barbará, and Carlotta Domeniconi. On-line LDA: adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 3–12, Washington, DC, USA, 2008. IEEE Computer Society. ISBN 978-0-7695-3502-9. doi: 10.1109/ICDM.2008.140.

[50] Tzu-Chuan Chou and Meng Chang Chen. Using incremental PLSI for threshold-resilient online event analysis. *IEEE Trans. on Knowl. and Data Eng.*, 20(3):289–299, 2008. ISSN 1041-4347. doi: http://dx.doi.org/10.1109/TKDE.2007.190702.

[51] S. Guha, A. Meyerson, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering data dtreams: theory and practice. *IEEE Trans. of Knowlende and Data Eng.*, 15 (3):515–528, 2003.

[52] Max Welling, Chaitanya Chemudugunta, and Nathan Sutter. Deterministic latent variable models and their pitfalls. In *SDM*, 2008. URL `http://www.ics.uci.edu/~welling/publications/papers/pVEM_v5.pdf`.

[53] Jure Ferlez, Christos Faloutsos, Jure Leskovec, Dunja Mladenic, and Marko Grobelnik. Monitoring network evolution using MDL. In *Proceedings of IEEE Int. Conf. on Data Engineering (ICDE'08)*. IEEE, 2008.

[54] P. Ipeirotis, A. Ntoulas, J. Cho, and L. Gravano. Modeling and managing content changes in text databases. In *Proceedings of the IEEE Int. Conf. on Data Engineering (ICDE'05)*, 2005.

[55] Wei Jin, Rohini K. Srihari, Hung Hay Ho, and Xin Wu. Improving knowledge discovery in document collections through combining text retrieval and link analysis techniques. In *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, pages 193–202, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-3018-4. doi: 10.1109/ICDM.2007.62.

[56] Chong Wang, David Blei, and David Heckerman. Continuous time dynamic topic models. In *Proceedings of ICML*, 2008.

[57] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 497–506, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-495-9. doi: http://doi.acm.org/10.1145/1557019.1557077.

[58] Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. Automatic labeling of multinomial topic models. In *KDD*, pages 490–499, 2007.

[59] Conrad Albrecht-Buehler, Benjamin Watson, and David A. Shamma. Visualizing live text streams using motion and temporal pooling. *IEEE Computer Graphics and Applications*, 25:52–59, 2005. ISSN 0272-1716. doi: http://doi.ieeecomputersociety.org/10.1109/MCG.2005.70.

[60] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. ThemeRiver: visualizing thematic changed in large document collections. *IEEE Trans. Visualization and Computer Graphics*, 8(1):9–20, 2002.

[61] P. Imrich, K. Mueller, D. Imre, A. Zelenyuk, and W. Zhu. 3D ThemeRiver. In *IEEE Information Visualization Symposium '03*, 2003.

[62] David Newman, Timothy Baldwin, Lawrence Cavedon, Eric Huang, Sarvnaz Karimi, David Martinez, Falk Scholer, and Justin Zobel. Visualizing search results and document collections using topic maps. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(2-3):169 – 175, 2010. ISSN 1570-8268. doi: DOI:10.1016/j.websem.2010.03.005.

[63] Qiaozhu Mei and ChengXiang Zhai. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 198–207, New York, NY, USA, 2005. ACM. ISBN 1-59593-135-X. doi: http://doi.acm.org/10.1145/1081870.1081895.

[64] W. S. Cleveland. *The elements of graphing data.* Hobart Press, Summit, New Jersey, U.S.A., 1985, 1994.

[65] Scott Golder and Bernardo Huberman. Usage patterns of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208, 2006.

[66] Fabian M. Suchanek, Milan Vojnovic, and Dinan Gunawardena. Social tags: Meanings and suggestions. In *CIKM'08*, pages 223–232, Napa Valley, CA, USA, Oct. 2008. ACM.

[67] Daniel Zeng and Huiqian Li. How useful are tags? An empirical analysis of collaborative tagging for web page recommendation. In Christopher Yang, Hsinchun Chen, Michael Chau, Kuiyu Chang, Sheau-Dong Lang, Patrick Chen, Raymond Hsieh, Daniel Zeng, Fei-Yue Wang, Kathleen Carley, Wenji Mao, and Justin Zhan, editors, *Intelligence and Security Informatics*, volume 5075 of *Lecture Notes in Computer Science*, pages 320–330. Springer Berlin / Heidelberg, 2008. ISBN 978-3-540-69136-5.

[68] Aleksandra Milicevic, Alexandros Nanopoulos, and Mirjana Ivanovic. Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions. *Artificial Intelligence Review*, 33:187–209, 2010. ISSN 0269-2821. doi: http://dx.doi.org/10.1007/s10462-009-9153-2.

[69] Eric Schwarzkopf, Dominik Heckmann, Dietmar Dengler, and Alexander Kröner. Mining the structure of tag spaces for user modeling. In *Data Mining for User Modeling Workshop at the UM'2007*, pages 63–75, Corfu, Greece, June 2007.

[70] Marco de Gemmis, Pasquale Lops, Giovanni Semeraro, and Pierpaolo Basile. Integrating tags in a semantic content-based recommender. In *RecSys'08*, pages 163–170, Lausanne, Switzerland, Oct. 2008. ACM.

[71] Xian Wu, Lei Zhang, and Yong Yu. Exploring social annotations for the semantic web. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 417–426, New York, NY, USA, 2006. ACM. ISBN 1-59593-323-9. doi: http://doi.acm.org/10.1145/1135777.1135839.

[72] Ding Zhou, Jiang Bian, Shuyi Zheng, Hongyuan Zha, and C. Lee Giles. Exploring social annotations for information retrieval. In *Proceeding of the 17th international*

conference on World Wide Web, WWW '08, pages 715–724, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-085-2. doi: http://doi.acm.org/10.1145/1367497.1367594.

[73] Xiance Si and Maosong Sun. Tag-LDA for scalable real-time tag recommendation. *Journal of Computational Information Systems*, 2009.

[74] Valentina Zanardi and Licia Capra. Social ranking: uncovering relevant content using tag-based recommender systems. In *Proceedings of the 2008 ACM conference on Recommender systems*, RecSys '08, pages 51–58, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-093-7. doi: http://doi.acm.org/10.1145/1454008.1454018.

[75] Pavan K. Vatturi, Werner Geyer, Casey Dugan, Michael Muller, and Beth Brownholtz. Tag-based filtering for personalized bookmark recommendations. In *CIKM'08*, pages 1395–1396, Napa Valley, CA, USA, Oct. 2008. ACM.

[76] Börkur Sigurbjörnsson and Roelof van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 327–336, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-085-2. doi: http://doi.acm.org/10.1145/1367497.1367542.

[77] Dominik Benz, Marko Grobelnik, Andreas Hotho, Robert Jäschke, Dunja Mladenic, Vito D. P. Servedio, Sergej Sizov, and Martin Szomszor. 08391 Working group summary – Analyzing tag semantics across tagging systems. In Harith Alani, Steffen Staab, and Gerd Stumme, editors, *Social Web Communities*, number 08391 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2008. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany. URL `http://drops.dagstuhl.de/opus/volltexte/2008/1785`.

[78] Paul Heymann and Hector Garcia-Molina. Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical report, InfoLab, Computer Science Department, Stanford University, April 2006.

[79] Takeharu Eda, Masatoshi Yoshikawa, Toshio Uchiyama, and Tadasu Uchiyama. The effectiveness of latent semantic analysis for building up a bottom-up taxonomy from folksonomy tags. *World Wide Web*, 12:421–440, December 2009. ISSN 1386-145X. doi: http://dx.doi.org/10.1007/s11280-009-0069-1.

[80] Ciro Cattuto, Dominik Benz, Andreas Hotho, and Gerd Stumme. Semantic grounding of tag relatedness in social bookmarking systems. In *ISWC '08: Proceedings of the 7th International Conference on The Semantic Web*, pages 615–631, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 978-3-540-88563-4. doi: http://dx.doi.org/10.1007/978-3-540-88564-1_39.

[81] Benjamin Markines, Ciro Cattuto, Filippo Menczer, Dominik Benz, Andreas Hotho, and Gerd Stumme. Evaluating similarity measures for emergent semantics of social tagging. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 641–650, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-487-4. doi: http://doi.acm.org/10.1145/1526709.1526796.

[82] Neil Ireson and Fabio Ciravegna. Toponym resolution in social media. In Peter Patel-Schneider, Yue Pan, Pascal Hitzler, Peter Mika, Lei Zhang, Jeff Pan, Ian Horrocks, and Birte Glimm, editors, *The Semantic Web – ISWC 2010*, volume 6496 of *Lecture Notes in Computer Science*, pages 370–385. Springer Berlin / Heidelberg, 2010. ISBN 978-3-642-17745-3.

[83] Maurizio Tesconi, Francesco Ronzano, Andrea Marchetti, and Salvatore Minutoli. Semantify del.icio.us: automatically turn your tags into senses. In *Proceedings of the First Social Data on the Web Workshop (SDoW2008)*, 2008.

[84] Ching-Man Au Yeung, Nicholas Gibbins, and Nigel Shadbolt. Tag meaning disambiguation through analysis of tripartite structure of folksonomies. In *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Workshops*, WI-IATW '07, pages 3–6, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-3028-1.

[85] Pasquale De Meo, Giovanni Quattrone, and Domenico Ursino. Exploitation of semantic relationships and hierarchical data structures to support a user in his annotation and browsing activities in folksonomies. *Information Systems*, 34(6): 511 – 535, 2009. ISSN 0306-4379. doi: DOI:10.1016/j.is.2009.02.004.

[86] Y. Hassan-Montero and V. Herrero-Solana. Improving Tag-Clouds as visual information retrieval interfaces. *Merida, InSciT2006 conference*, 2006.

[87] Emily Moxley, Jim Kleban, Jiejun Xu, and B. S. Manjunath. Not all tags are created equal: learning flickr tag semantics for global annotation. In *ICME'09: Proceedings of the 2009 IEEE international conference on Multimedia and Expo*, pages 1452–1455, Piscataway, NJ, USA, 2009. IEEE Press. ISBN 978-1-4244-4290-4.

[88] Ricardo Usbeck. Analyse divergenten Nutzerverhaltens in kollaborativen Tagging Systemen. Bachelor thesis, Martin Luther University Halle-Wittenberg, Halle, Germany, 2010.

[89] Júlio S.L.T. Militao, Vicente P. Emerenciano, Marcelo J.P. Ferreira, Daniel Cabrol-Bass, and Michel Rouillard. Structure validation in computer-supported structure elucidation: 13C NMR shift predictions for steroids. *Chemometrics and Intelligent Laboratory Systems*, 67(1):5 – 20, 2003. ISSN 0169-7439. doi: 10.1016/S0169-7439(03)00057-1.

[90] Dionisia Sanz, Rosa M. Claramunt, Anil Saini, Vinod Kumar, Ranjana Aggarwal, Shiv P. Singh, Ibon Alkorta, and José Elguero. Pyrazolo[1,5-a]pyrimidines. a combined multinuclear magnetic resonance (1H, 13C, 15N, 19F) and DFT approach to their structural assignment. *Magnetic Resonance in Chemistry*, 45 (6):513–517, 2007. ISSN 1097-458X. doi: 10.1002/mrc.1992. URL `http://dx.doi.org/10.1002/mrc.1992`.

[91] Mikhail Elyashberg, Kirill Blinov, and Antony Williams. A systematic approach for the generation and verification of structural hypotheses. *Magnetic Resonance in Chemistry*, 47(5):371–389, 2009. ISSN 1097-458X. doi: 10.1002/mrc.2397. URL `http://dx.doi.org/10.1002/mrc.2397`.

[92] Kyle L. Jensen, Abigail S. Barber, and Gary W. Small. Simulation of carbon-13 nuclear magnetic resonance spectra of polycyclic aromatic compounds. *Analytical Chemistry*, 63(11):1081–1090, 1991. doi: 10.1021/ac00011a007. URL `http://pubs.acs.org/doi/abs/10.1021/ac00011a007`.

[93] Vladimir Kvasnicka, Stepan Sklenak, and Jiri Pospichal. Application of recurrent neural networks in chemistry. Prediction and classification of carbon-13 NMR chemical shifts in a series of monosubstituted benzenes. *Journal of Chemical Information and Computer Sciences*, 32(6):742–747, 1992. doi: 10.1021/ci00010a023.

[94] Daniel Svozil, Jiri Pospichal, and Vladimir Kvasnicka. Neural network prediction of carbon-13 NMR chemical shifts of alkanes. *Journal of Chemical Information and Computer Sciences*, 35(5):924–928, 1995. doi: 10.1021/ci00027a021. URL `http://pubs.acs.org/doi/abs/10.1021/ci00027a021`.

[95] Deborah L. Clouser and Peter C. Jurs. Simulation of the 13C nuclear magnetic resonance spectra of ribonucleosides using multiple linear regression analysis and neural networks. *Journal of Chemical Information and Computer Sciences*, 36(2): 168–172, 1996. doi: 10.1021/ci950055y. URL `http://pubs.acs.org/doi/abs/10.1021/ci950055y`.

[96] Shushen Liu, Hailing Liu, Banmei Yu, Chenzhong Cao, and Shengshi Zhiliang Li. Investigation on quantitative relationship between chemical shift of carbon-13 nuclear magnetic resonance spectra and molecular topological structure based on a novel atomic distance-edge vector (adev). *Journal of Chemometrics*, 15(5):427–438, 2001. ISSN 1099-128X. doi: 10.1002/cem.632.

[97] Ernesto Fattorusso, Paolo Luciano, Adriana Romano, Orazio Taglialatela-Scafati, Giovanni Appendino, Marianna Borriello, and Caterina Fattorusso. Stereostructure assignment of medium-sized rings through an NMR-computational combined approach. Application to the new germacranes ketopelenolides c and d. *Journal of Natural Products*, 71(12):1988–1992, 2008. doi: 10.1021/np8003547.

[98] Lawrence S. Anker and Peter C. Jurs. Prediction of carbon-13 nuclear magnetic resonance chemical shifts by artificial neural networks. *Analytical Chemistry*, 64 (10):1157–1164, 1992. doi: 10.1021/ac00034a015.

[99] Brooke E. Mitchell and Peter C. Jurs. Computer assisted simulation of 13C nuclear magnetic spectra of monosaccharides. *Journal of Chemical Information and Computer Sciences*, 36(1):58–64, 1996. doi: 10.1021/ci950262y.

[100] M.E. Elyashberg, A.J. Williams, and G.E. Martin. Computer-assisted structure verification and elucidation tools in NMR-based structure elucidation. *Progress in Nuclear Magnetic Resonance Spectroscopy*, 53(1-2):1 – 104, 2008. ISSN 0079-6565. doi: 10.1016/j.pnmrs.2007.04.003.

[101] K A Blinov, M E Elyashberg, S G Molodtsov, A J Williams, and E R Martirosian. An expert system for automated structure elucidation utilizing 1H-1H, 13C-1H and 15N-1H 2D NMR correlations. *Fresenius Journal Of Analytical Chemistry*, 369(7-8):709–714, 2001.

[102] Christie Bradley D. and Munk Morton E. The application of two-dimensional nuclear magnetic resonance spectroscopy in computer-assisted structure elucidation. *Analytica Chimica Acta*, 200(0):347 – 361, 1987. ISSN 0003-2670. doi: 10.1016/S0003-2670(00)83782-4.

[103] A. Korytko, K.-P. Schulz, M. S. Madison, and M. E. Munk. HOUDINI: a new approach to computer-based structure generation. *Journal of Chemical Information and Computer Sciences*, 43(5):1434–1446, 2003. doi: 10.1021/ci034057r.

[104] Christoph Steinbeck. SENECA: A platform-independent, distributed, and parallel system for computer-assisted structure elucidation in organic chemistry. *Journal of Chemical Information and Computer Sciences*, 41(6):1500–1507, 2001. doi: 10.1021/ci000407n.

[105] Yongquan Han and Christoph Steinbeck. Evolutionary-algorithm-based strategy for computer-assisted structure elucidation. *Journal of Chemical Information and Computer Sciences*, 44(2):489–498, 2004. doi: 10.1021/ci034132y.

[106] Jean-Marc Nuzillard and Massiot Georges. Logic for structure determination. *Tetrahedron*, 47(22):3655 – 3664, 1991. ISSN 0040-4020. doi: 10.1016/S0040-4020(01)80878-4.

[107] Michaela Hohenner, Sven Wachsmuth, and Gerhard Sagerer. Modelling expertise for structure elucidation in organic chemistry using bayesian networks. *Knowledge-Based Systems*, 18(4-5):207 – 215, 2005. ISSN 0950-7051. doi: 10.1016/j.knosys.2005.03.001.

[108] Elena Erosheva Stephen, Stephen Fienberg, and John Lafferty. Mixed membership models of scientific publications. In *Proceedings of the National Academy of Sciences*, pages 5220–5227. press, 2004. doi: doi:10.1073/pnas.0307760101.

[109] David Mimno, Hanna M. Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2*, EMNLP '09, pages 880–889, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-62-6.

[110] Xiaochuan Ni, Jian-Tao Sun, Jian Hu, and Zheng Chen. Mining multilingual topics from wikipedia. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 1155–1156, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-487-4. doi: http://doi.acm.org/10.1145/1526709.1526904.

[111] Bing Zhao and Eric P. Xing. BiTAM: bilingual topic admixture models for word alignment. In *In Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics*, 2006.

[112] Bing Zhao and Eric P. Xing. HM-BiTAM: bilingual topic exploration, word alignment, and translation. In *Advances in Neural Information Processing Systems*, 2007.

[113] Yik-Cheung Tam, Ian Lane, and Tanja Schultz. Bilingual LSA-based adaptation for statistical machine translation. *Machine Translation*, 21:187–207, December 2007. ISSN 0922-6567. doi: 10.1007/s10590-008-9045-2.

[114] John C. Platt, Kristina Toutanova, and Wen-Tau Yih. Translingual document representations from discriminative projections. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 251–261, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[115] Joao V. Graca, Kuzman Ganchev, and Ben Taskar. Expectation maximization and posterior constraints. In *Advances in Neural Information Processing Systems*, volume 20, 2007.

[116] 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. ISSN 1532-4435.

[117] Jordan Boyd-Graber and David M. Blei. Multilingual topic models for unaligned text. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 75–82, Arlington, Virginia, United States, 2009. AUAI Press. ISBN 978-0-9749039-5-8.

[118] Jagadeesh Jagaralamudi and Hal Daumé. Extracting multilingual topics from unaligned corpora. In *Proceedings of the European Conference on Information Retrieval (ECIR)*, Milton Keynes, United Kingdom, 2010.

[119] Duo Zhang, Qiaozhu Mei, and ChengXiang Zhai. Cross-lingual latent topic extraction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1128–1137, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[120] Ramaswamy Nilakantan, Norman Bauman, J. Scott Dixon, and R. Venkataraghavan. Topological torsion: a new molecular descriptor for SAR applications. Comparison with other descriptors. *J Chem Inf Comp Sci.*, 27(2):82 et sqq., 1987.

[121] James Keeler. *Understanding NMR Spectroscopy.* John Wiley & Sons, Ltd., 2005. ISBN 0470017872.

[122] J. Shoolery. *A basic guide to NMR.* Varian Associates, Palo Alto, CA, 1972.

[123] R. Freeman. *Magnetic resonance in chemistry and medicine.* Oxford University Press, New York, USA, 2003.

Lebenslauf

Persönliche Daten

Name:	André Gohr
Geburtsdatum:	13. Oktober 1979
Geburtsort:	Potsdam
Staatsangehörigkeit:	deutsch
Familenstand:	ledig

Schulbildung

1992–1998:	Werner-von-Siemens Gymnasium Magdeburg
1998:	Abschluß mit dem Abitur

Universitäre Bildung

01.10.1999–30.09.2000:	Studium der Biotechnologie an der Technischen Universität Braunschweig
01.10.2001:	Studium der Bioinformatik an der Martin-Luther Universität Halle-Wittenberg
07.04.2006 :	Diplom der Bioinformatik an der Martin-Luther Universität Halle-Wittenberg

Berufliche Tätigkeiten

14.04.2006–15.04.2007	wiss. Mittarbeiter am Lehrstuhl für Bioinformatik und Mustererkennung (Prof. Posch) der Martin-Luther Universität Halle-Wittenberg
15.04.2007–30.09.2010	wiss. Mittarbeiter in der Abteilung Natur- und Wirkstoffchemie (Prof. Wessjohann) des Leibniz-Instituts für Pflanzenbiochemie Halle
seit 01.10.2010	wiss. Mittarbeiter am Lehrstuhl für Bioinformatik (Prof. Große) der Martin-Luther Universität Halle-Wittenberg

Halle, den 15.05.2012 André Gohr

Erklärung

Hiermit erkläre ich, dass ich diese Arbeit selbstständig und ohne fremde Hilfe verfasst habe. Ich habe keine anderen als die von mir angegebenen Quellen oder Hilfsmittel verwendet. Wörtliche und sinngemäße Zitate habe ich als solche kenntlich gemacht. Ich habe mich bisher nicht um den Doktorgrad beworben.

Halle, den 15.05.2012 André Gohr