Graph-Based Recommendation in Broad Folksonomies

vorgelegt von
Diplom-Ingenieur
Robert Wetzker

Von der Fakultät IV – Elektrotechnik und Informatik –
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
– Dr.-Ing. –

genehmigte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr. Möller
Berichter: Prof. Dr. Albayrak
Berichter: Prof. Dr. Bauckhage

Tag der wissenschaftlichen Aussprache: 14. Mai 2010

Berlin 2010

D 83

Abstract

The user-centric annotation of resources by freely chosen words (tags) has become the predominant form of content categorization of the Web 2.0 age. It provided the foundations for the success of now famous services, such as Delicious, Flickr, LibraryThing, or Last.fm. Tags have proven to be a powerful alternative to existing top-down categorization techniques, as for example taxonomies or predefined dictionaries, which lack flexibility and are expensive in their creation and maintenance. Instead, tagging allows users to choose the labels that match their real needs, tastes, or language, which reduces the required cognitive effort.

Services that center on the collaborative tagging of resources are called folksonomies. These communities have become an invaluable source for information retrieval, since they bundle the interests of thousand or even millions of users, magnifying the underlying domain from a user-centric perspective. Furthermore, the existence of tags allows for the unified modeling of content independent of its type or format and without the need of expensive content retrieval, processing, storage, or indexing. Of special interest from an information retrieval perspective are the tags that are generated in "broad folksonomies", where many users reference and tag the same objects. This collaborative tagging produces high quality content descriptors and characteristic tag spectra with high potential for better content models.

This dissertation aims to develop novel algorithms for the personalized recommendation of objects within folksonomies. Recommender systems support users in the discovery of valuable content out of an overwhelming set of choices and already increase the perceived value of many web applications. However, the tripartite structure of folksonomies that results from the interaction of content, tags, and users goes beyond the capability of common recommendation techniques. The algorithms presented in this dissertation are instead especially tailored towards to needs of folksonomies. They incorporate tagging and usage patterns in parallel in order to derive more precise and robust recommendation models. Exhaustive tests for different settings and various folksonomies show that the developed solutions produce more

accurate recommendations than the current *state-of-the-art*.

Zusammenfassung

Die nutzerzentrierte Klassifikation von Ressourcen durch Zuweisung frei wählbarer Wörter (Tags) hat sich zu einem integralen Bestandteil moderner Webanwendungen entwickelt. Während sich kontrollierte Klassifikationsschemata wie etwa Taxonomien bzw. Ontologien aus Nutzersicht häufig als zu starr, kostspielig und wenig dynamisch erweisen, gewähren Tags ein hohes Maß an Flexibilität und erfordern nur geringen kognitiven Aufwand während des Klassifikationssprozesses. Die Nutzerfreundlichkeit und Effektivität dieser Art der unkontrollierten Inhaltsklassifikation begründete den Erfolg und die heutige weite Verbreitung von Tagging Communities wie Delicious, Flickr, LibraryThing oder Last.fm, welche man auch als Folksonomien bezeichnet. Folksonomien bündeln die Interessen von Tausenden oder Millionen von Nutzern und werden dadurch zu einzigartigen Informationsquellen. Die Existenz von Tags erlaubt darüber hinaus eine einheitliche Modellierung von Inhalten unabhängig von ihrer Art oder ihres Formats und erspart in vielen Fällen eine aufwendige Verarbeitung, Speicherung sowie Indizierung der Inhalte selbst. Aus Datenverarbeitungssicht besonders interessant sind sogenannte "broad folksonomies", bei welchen die selben Inhalte von mehreren Nutzern referenziert und annotiert werden können, so dass es zur Bildung deskriptiver und charakteristischer Worthäufungen kommt.

In dieser Dissertation werden Algorithmen zur personalisierten Empfehlung von Inhalten in Folksonomien entworfen und evaluiert. Empfehlungssysteme unterstützen Nutzer beim Entdecken interessanter Inhalte aus einer unübersichtlichen Menge existierender Möglichkeiten. Obwohl bereits umfangreiche Forschungsarbeiten auf dem Gebiet von Empfehlungssystemen existieren, lassen sich diese nicht ohne Informationsverlust auf die tripartite Folksonomiestruktur aus Inhalten, Nutzern und Tags übertragen. Die in dieser Arbeit entwickelten Algorithmen berücksichtigen hingegen diese besondere Graphstruktur und profitieren von den durch Tags gegebenen semantischen Zusatzinformationen. Ausführliche Tests in verschiedenen Szenarien und für mehrere Folksonomien zeigen, dass die entwickelten Lösungen bessere Empfehlungen liefern als existierende Ansätze.

Acknowledgments

This dissertation would not have been possible without the constant encouragement and support of many inspiring people.

I was fortunate to have two experienced supervisors during my time at the DAI-Labor of the Technische Universität Berlin. Professor Sahin Albayrak convinced me to follow the academic career path and provided the guidance as well as the freedom that were essential for this dissertation. Professor Christian Bauckhage introduced me to the world of research and has been an enduring source of inspiration and assistance ever since.

Furthermore, I have been lucky to work with an amazing group of researchers during the past four years. I am deeply thankful to: Tansu Alpcan, Leonhard Hennig, Alexander Korth, Florian Metze, Nicolas Neubauer, Till Plumbaum, Alan Said, Winfried Umbrath, and Carsten Zimmermann.

I would especially like to thank Tansu Alpcan, Florian Metze, and Christian Bauckhage who introduced me to the topic of folksonomies and provided the opportunity for extensive study in this direction. I thank Winfried Umbrath and Alan Said for all the fruitful discussions and countless MATLAB nights. I also want to thank Jérôme Kunegis, who constantly challenged the problems and solutions we discussed. Trying to convince him of my ideas always contested previous believes and led to new insights. I further thank all other members of the competence center on information retrieval and machine learning (CC-IRML) as well as the entire team of the DAI-Labor for their collaboration and support during the last years.

Finally, I would like to express my deep gratitude to the many people who proofread this dissertation: Leonhard Hennig, Alan Said, Andreas Hotho, Alice Motanga, Carsten Zimmermann, Stefan Fricke, and Jérôme Kunegis. Their valuable comments, questions, and ideas have helped to improve the quality and coherence of this dissertation significantly.

Last but not least I thank my family. For everything.

Publications

Many of the materials and concepts in this thesis have appeared in previous publications by the author.

- Robert Wetzker, Carsten Zimmermann, Christian Bauckhage, Sahin Albayrak, *I tag, You tag: Translating Tags for Advanced User Models*, In: 3rd ACM Int. Conf. on Web Search and Data Mining (WSDM), 2010

- Alan Said, Robert Wetzker, Winfried Umbrath, Leonhard Hennig, *A hybrid PLSA approach for warmer cold start in folksonomy recommendation*, In: 3rd ACM Conf. on Recommender Systems, Workshop on Recommender Systems and the Social Web, 2009

- Robert Wetzker, Alan Said, Carsten Zimmermann, *Understanding the user: Personomy translation for tag recommendation*, In: Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD), Discovery Challenge, 2009

- Nicolas Neubauer, Robert Wetzker, Klaus Obermayer, *Tag Spam Creates Large Non-Giant Connected Components*, In: 18th Int. World Wide Web Conference, 5th Int. Workshop on Adversarial Information Retrieval on the Web (AIRWEB), 2009

- Robert Wetzker, Winfried Umbrath, Alan Said, *A hybrid approach to item recommendation in folksonomies* (Best Technical Paper), In: 2nd ACM Int. Conf. on Web Search and Data Mining (WSDM), Workshop on Exploiting Semantic Annotations in Information Retrieval (ESAIR), 2009

- Robert Wetzker, Carsten Zimmermann, Christian Bauckhage, *Detecting Trends in Social Bookmarking Systems: A del.icio.us Endeavor*, In: Int. Journal of Data Warehousing and Mining (to appear)

- Christian Bauckhage, Tansu Alpcan, Robert Wetzker, Winfried Umbrath, *IMAGE RETRIEVAL AND WEB 2.0: WHERE CAN WE GO FROM HERE?*, In: 15th IEEE Int. Conf. on Image Processing (ICIP), 2008

- Robert Wetzker, Till Plumbaum, Alexander Korth, Christian Bauckhage, Tansu Alpcan, Florian Metze, *Detecting Trends in Social Bookmarking Systems using a Probabilistic Generative Model and Smoothing*, In: Int. Conf. on Pattern Recognition (ICPR), 2008

- Robert Wetzker, Carsten Zimmermann, Christian Bauckhage, *Analyzing Social Bookmarking Systems: A del.icio.us Cookbook*, In: Eur. Conf. on Artificial Intelligence (ECAI), 2008

Contents

List of Figures

List of Tables

Introduction

Motivation

Tagging has become a ubiquitous form of content classification on the web. The labeling of content with sets of freely-chosen keywords (tags) has proven to be a powerful alternative to existing top-down categorization techniques, such as dictionaries or taxonomies that lack flexibility and are often expensive in their creation and maintenance. Tagging outsources these efforts to the community. Furthermore, the openness of tagging better matches a user's real needs and language [Qui05], and the free assignment of words requires little cognitive costs and lowers the entry barriers for new users [Mat04]. The flexibility of tagging makes it a powerful tool for organizing collections of any content type, including websites, movies, books, and publications. Not surprisingly, we find tagging portals, such as Delicious[1] and Flickr[2], among the stars of the Web 2.0 era.

Community services that center around tagging have been branded **folksonomies**, a composition of the terms "folk" and "taxonomy" [Wal04]. The success of folksonomies originated from members' ability to centrally collect and manage content collections on the web, overcoming local storage policies. Users of folksonomies are granted ubiquitous access to their collections of photos, websites, or publications independent of their current location or the currently used device. These personal advantages are complemented by a social component. This derives from the fact that most folksonomies allow the sharing of content with other users, the discovery of content other users considered interesting, or the communication with other users through various channels. Folksonomies come in different forms with one distinguishing criterion being the tagging rights. **Narrow folksonomies**, such as Flickr or Technorati[3], grant tagging rights to the owners of a photo or the author of a blog post exclusively. This results in only few labels per resource that are

[1] http://delicious.com
[2] http://www.flickr.com
[3] http://technorati.com

biased towards the tagging habits of one user. **Broad folksonomies**, on the other hand, allow *free-for-all tagging* [MNBD06] where any user can tag any resource. This collaborative tagging activity boosts the semantic quality of labels, since user-level discrepancies in tagging behavior are compensated by accumulative effects [Sur04]. The labeling of items by many users leads to the emergence of item-specific tag spectra that can help in the development of more robust item models [GH06].

Aggregating the interests of potentially millions of users, broad folksonomies become a magnifier of the underlying domain. The popularity of high traffic websites will reappear in a bookmarking service such as Delicious [HKGM08], and influential publications will also gain popularity in research folksonomies like CiteULike[4] or Bibsonomy[5]. Recent studies further report that events and trends within the underlying domain, such as the advent of a new website or a new influential blog entry, tend to echo within a folksonomy without much delay [HKGM08]. The access to this *collective knowledge* comes at a low cost, since all data is centrally available and no continuous, distributed crawling as for web indexing is required. Instead, we find that the meta-data provided by tags can substitute the inefficient processing of resources. This is especially compelling when folksonomies aggregate different content types, e.g. HTML, video, PDF, etc.. In such heterogeneous environments, content-based item models are rather inappropriate, since they require the implementation of different processing techniques. Furthermore, they face the non-trivial task of modeling all content types in a uniform way to perform comprehensive similarity calculations. The description of content by tags overcomes these limitations and permits a unified modeling independent of resource-level heterogeneities. All these characteristics make folksonomies an invaluable source for next generation (web) search, recommendation, and trend indication solutions.

Recommender systems help users to discover items of interest from a potentially overwhelming set of choices [RV97] and have become an integral part of many services on the web. In order to provide valuable suggestions, they have to predict future user interests and to identify relevant items based on the available data. Recommender systems can be designed in multiple ways with the most common approaches being content-based and collaborative filtering models. Content-based models derive object similarities based on their content and recommend those objects that are similar to the objects a user was previously interested in. On the other hand, collaborative filtering methods look at usage patterns and recommend objects

[4] http://www.citeulike.org

[5] http://www.bibsonomy.org

that users with similar tastes liked in the past. Yet another class of recommender systems, so called hybrid recommenders, combines content-based and collaborative techniques to overcome the deficiencies of each separate method. However, folksonomies are generally modeled as a tripartite hypergraph that reflects the co-occurrence patterns between users, items, and tags. This hypergraph cannot be directly approached by standard recommendation principles. A naïve strategy for applying existing techniques to folksonomies could ignore one of the three object types and build recommendation models from the resulting bipartite co-occurrence patterns of items and tags, items and users, or users and tags. These bipartite models would then resemble common collaborative filtering settings that work on item-user co-occurrence observations or content-based settings, where tags act as content descriptors. However, at least for broad folksonomies, this reduction of the folksonomy hypergraph is accompanied by information loss, lowering the potential quality of recommendations. Accordingly, previous solutions that tried to reflect the hypergraph nature of folksonomies during the process of model-building were reported to produce more accurate suggestions than naïve methods [WUS09, HJSS06a, SNM08, AS08].

One recurrent observation is that user-generated annotations, i.e. tags, can contribute to the design of more effective recommendation engines, since they deliver insights about the intentions of users and characterize the nature of items. Tags act as essential connectors between items and users and thus help to reduce the data sparsity in classical collaborative filtering scenarios. The interaction between users, items, and tags is still not well understood, especially with respect to potential recommendation scenarios. Methods that respect the hypergraph nature of folksonomies and integrate tags into a unified recommendation model are still rare and leave much room for sophistication.

Contributions and structure

This dissertation develops novel recommendation models that are specifically tailored to folksonomies. The design of effective recommender systems requires deep knowledge about the underlying domain. One of the major contributions of this work is therefore an extensive analysis of structural and temporal aspects found in real-world collaborative tagging communities and the estimation of their potential impact on recommendation. The analytical contributions are presented in **Part I** of this dissertation and consist of:

Large-scale corpus retrieval. Our analyses are mainly based on a large-scale snapshot of the Delicious social bookmarking service crawled between December 2007, and April 2008. The final corpus consists of 132 million Delicious bookmarks from around 950.000 users, which, to the best of our knowledge, makes it the biggest folksonomy dataset that has been analyzed in academia so far. The sheer size of this data and its relative completeness enables analyses that have not been possible before. The public release of the entire dataset to assist the research activities of other groups is an additional contribution of this work. (→ **Section** 2.1)

Analysis of real-world folksonomies. This work presents a comprehensive analysis of the structural and temporal patterns within Delicious and a variety of other broad folksonomies. First, it presents an exhaustive study of the Delicious corpus (→ **Chapter** 2) which elaborates on what content Delicious users actually bookmark, how this bookmarking behavior changes over time, and to what degree interests are shared among members of the community (→ **Section** 2.2). Moreover, it investigates the motivations behind tagging and presents a profound analysis of the tag distributions that emerge for users and items over time (→ **Section** 2.3). The following chapter then discusses folksonomies from a network perspective. The chapter investigates global network properties and the evolution of these properties when folksonomies start to grow (→ **Chapter** 3). All findings are discussed with respect to their effect on potential recommendation scenarios and their significance for the design of effective folksonomy recommenders.

Spam detection. An initial analysis of the Delicious corpus showed that the data was considerably polluted by spam. Similar to other social media, such as blogs or email, spam seems to be a common phenomenon in folksonomies as well. This spam needs to be filtered or suppressed before any descriptive recommendation model can be built. This dissertation therefore investigates the intentions behind spamming and their impact on the data. It then presents two novel approaches to spam handling. The first approach detects spam based on its connectivity patterns within the folksonomy graph. The second method suppresses spam by moving from an occurrence-based perspective to a diffusion-based one. We then study the effect of spam filtering on different aspects of the Delicious dataset. (→ **Chapter** 4)

Discussion of related work I. The first part of this dissertation concludes with an exhaustive discussion of previously presented work related to folksonomies and folksonomy analysis and emphasizes the novel contributions of this dissertation. (→ **Chapter** 5)

Based on the analytical findings of the first part, **Part II** of this dissertation is dedicated to the design of novel recommendation solutions for folksonomies. It starts with an introduction to the recommendation problem from a folksonomy perspective and an overview of potential application areas (→ **Chapter** 6). The main contribution of the second part of this dissertation is the introduction of three novel approaches that advance the *state-of-the-art* in the most relevant folksonomy recommendation tasks:

HyPLSA. The hybrid PLSA model presented in this work extends the well known probabilistic latent semantic analysis (PLSA) to a folksonomy scenario and merges tag data and usage patterns into a single recommendation model. This allows us to model users by the topics they are interested in, items by the topics they cover, and tags by the topics they represent. Inferring all these topics from the same topic space helps us overcome problems related to data sparsity and to derive more accurate recommendations in various settings. (→ **Chapter** 7)

UCTM. Going one step further, the user-centric tag model avoids the decomposition of the folksonomy network and works directly on the graph between users, items, and tags instead. The model is motivated by one striking observation of the preceding analysis, namely that users develop their individual tag vocabularies, so called "personomies" [HJSS06a], over time. This heterogeneity impedes direct comparisons of users or between users and items in the tag space. Our model accounts for these user-level differences in tagging by translating all user tags to a global tag vocabulary. Evaluation results show that this translation helps to infer the meaning of user tags, leading to improved accuracy in tag recommendation and tag-based search scenarios. (→ **Chapter** 8)

Tag-based trend detection. Recommending interesting objects at an early stage is an important recommendation task. This dissertation therefore presents a novel generative probabilistic model for the detection of new and seasonal trends in a folksonomy. The trend detector is built on sound statistical foundations and uses distributional smoothing in order to tune the trade-off between relative and absolute trends. (→ **Chapter** 9)

Discussion of related work II. The second part of this dissertation concludes with a discussion of previously proposed solutions in the field of folksnomy recommendation together with a review of general recommendation approaches that are closely related to this work's contributions. (→ **Chapter** 10)

A comparison of all novel approaches to the *state-of-the-art* in the respective domains shows improvements in model descriptiveness and in the quality of produced recommendations.

Application scenarios

Many potential recommendation scenarios materialize in a folksonomy setting[6]. This dissertation will mainly focus on the identification of relevant resources with respect to a given context. Here, context will most often represent a user with individual information needs but may also refer to a set of user-provided tags, e.g. in a search scenario. Both of these settings extend the basic item recommendation task and represent the recommendation scenarios of highest practical relevance. In addition to item recommendation this dissertation will present novel solutions to two other application scenarios: tag recommendation and tag-based trend detection. Tag recommendation is probably the best studied recommendation scenario in a folksonomy context. Including this scenario will therefore allow us to compare the quality of our solutions to existing approaches. On the other hand, the exploitation of folksonomies for trend detection is a rather novel research area with a strong focus on temporal data patterns. Tag-based trend detectors provide users with on-time recommendations and may help to narrow the gap between recommender systems and the real-time web. A short description of all application scenarios is given below:

Personalized item recommendation. Folksonomies have become rich sources of information. The challenge in a personalized item recommendation scenario is to predict those items, i.e. websites, publications, movies etc., a user will most likely find interesting. To address this task it is necessary to identify a user's previous interest patterns and to determine novel items that are likely to fall within these interests.

Tag-based social search. The tag-based social search scenario considers a user who wants to discover interesting items related to a given tag. The challenge in this scenario is not only the personalized retrieval of interesting items, but also the identification of the search intention behind the query tag. Even though the discussion is limited to a search scenario, our findings are also highly relevant for related tasks such as the topic-aware recommendation of items or the topic-aware detection and indication of trends.

[6] see Chapter 6

Tag recommendation. Tag recommenders support a user during the posting process by suggesting tags she may want to assign. In addition to their positive effect on usability, they are effective tools to limit global and local tag divergence by increasing the likelihood of tag reassignments and lowering the ratio of misspellings. Tag recommendation is probably the best-researched recommendation scenario in the context of folksonomies. Including this scenario therefore allows for the evaluation of the presented tag models with respect to a variety of previously proposed methods.

Tag-based trend detection. The final scenario investigates the task of recommending upcoming objects within a folksonomy. This requires a deep understanding of the temporal dynamics within folksonomies. Users benefit from early recommendations, since these provide insights about beginning shifts in community interests and may point a user to topics she has not yet been aware of. Though our focus will lie on the detection of tag-based trends, the generic design of the proposed trend detector also makes it applicable to detect trends in the item or user dimensions.

Related publications

Various contributions of this dissertation have been previously published in the form of journal articles or have been presented during conferences. Table 1 lists these contributions and the chapters in which they appear.

Chapter 2	[WZB08], [WZB10a], [WZB10b]
Chapter 4	[WZB08], [WZB10a], [NWO09]
Chapter 7	[WUS09], [SWUH09]
Chapter 8	[WSZ09], [WZB10b]
Chapter 9	[WPK+08], [WZB10a]

Table 1: Related publications to which the author contributed and their appearance throughout this dissertation.

Out of scope

There also exist various related research topics that, despite their importance, cannot be investigated here. These include:

Diversification. Many scenarios may require the diversification of suggestions and a low redundancy among the recommended objects [vZMGPR08]. Throughout this work, we will only consider different forms of accuracy as quality measures and ignore diversity.

External knowledge. This dissertation studies graph-based scenarios, where recommenders are built from the tripartite hypergraph of users, items, and tags. Information stemming from external sources, such as the actual content of an item [RHMGM09] or (meta-) data about users and items [TSD08] is not considered. Furthermore, we disregard external grammars, such as WordNet[7], for the enrichment of our tag models [AYGS09]. The solutions presented in this dissertation can thus be thought of as core building blocks for successful recommendation models, which can be enriched with additional external knowledge on demand.

Recommendation as process. Recommender systems may allow users to interact with the result set in order to find interesting items [MR09]. Recommendation then needs to be seen from a process perspective, and evaluation metrics would focus on process-oriented measures, such as the number of interactions required to find an item or the overall process time. This work will only investigate non-iterative recommendation scenarios instead.

Service specific structures. Most folksonomies offer additional functionality, such as the forwarding of bookmarks to friends or groups, the creation of meta tags, the commenting on other user's content, or the contribution of descriptions or bibliographic data for items. Information from these sources can improve user and item models and thus recommendation quality, e.g. [ABB+09]. However, feature sets among folksonomies differ in nature and scale. In order to obtain universally valid results, this dissertation therefore only considers the core folksonomy graph of all services. Cases where patterns within this graph are caused by service specific settings will be emphasized.

User studies. User studies are probably the most accurate way to evaluate recommender systems. However, in accordance with most work on recommender systems, we will simulate user studies by assuming that users have expressed their interest in an item by bookmarking it. Items are likewise considered as uninteresting for a user if they have not been bookmarked.

[7] `http://wordnet.princeton.edu`

The first part of this work investigates structural and temporal patterns that can be found in folksonomies and discusses their potential impact on the design of folksonomy recommenders. Part II then builds on these analytical findings and presents novel recommendation algorithms that are especially designed for folksonomies.

Part I

The Nature of Folksonomies

Chapter 1

An introduction to folksonomies

The functionality to save web pages (URLs) for later retrieval has been part of web browsers since the Mosaic browser in 1993[1]. Bookmarks have since then become a standard feature of all common web browsers. The necessity to catalog and retrieve previously visited and saved websites also led to the advent of social bookmarking services, such as Delicious, StumbleUpon[2], or CiteULike. The success of these services originated from members' ability to centrally store and manage bookmarks on the web, overcoming the limitations of browser-centric, local storage policies. Nowadays, social bookmarking goes well beyond the storage of interesting web links, and various bookmarking services guarantee users the ubiquitous access to their collections of favorite publications, songs, photos, or movies independent of the used browser or device.

One paradigm that paved the way to success of social bookmarking services is tagging. Tagging permits each user to describe and categorize books, photos, publications, websites, places, persons etc. with freely chosen keywords, so called "tags". Each user can choose the labels she considers most appropriate for a given content without being restricted to a predefined vocabulary. For instance, a user of Flickr who wants to organize photos of a trip to Paris could label a photo with "paris", "vacation", "2009", "art", "monalisa", "highqualityphoto", and "louvre". This will allow her to easily recover all photos of the year "2009" or to get a fast overview of photos related to "paris" and "art". In addition to descriptive tags, she is also free to assign tags, like "highqualityphoto", that support individual organizational

[1] http://groups.google.com/group/comp.infosystems/browse_thread/thread\/f39e59346ee6bc8d/563e36cb72f214d7

[2] http://www.stumbleupon.com/

tasks. Tagging is thus a categorization tool that suits the individual needs of every user, and users can choose those tags that best match their own language, background, or categorization strategy. This flexibility makes tagging a powerful alternative to previously predominant but inflexible categorization techniques, such as taxonomies or controlled category sets. In consequence, tagging has become a de facto standard in environments where content is tagged by non-expert users.

Tagging can be a powerful tool for the personal management of files[3], publications[4], or emails[5]. However, from the perspective of information retrieval, tags become especially appealing in social environments, where the collective tagging activity of many users is bundled in a single service. Their tagging support made social bookmarking services one of the first collaborative tagging communities, so called "folksonomies" [Wal04]. These communities aggregate the preferences, contributions, or views of their user base, turning them into impressive sources of information. With the overall number of collected URLs going into the hundreds of millions, social bookmarking services, like Delicious, already provide an excellent alternative to search engines for the discovery of relevant web content. Other folksonomies, such as Flickr or YouTube, have become a unique source to look for interesting photos or videos. The research folksonomies Bibsonomy and CiteUlike substitute libraries and facilitate the exploration of publications of every type and research field. In all these scenarios and independent of the respective content type, we find tags to act as connectors between users who seek information and the relevant content. Although the motivations behind tagging are primarily personal, our studies show the collaborative tagging of resources by many users to produce high quality content descriptors despite user-level differences in tagging behavior [Sur04]. These semantics that emerge despite the openness of tagging make folksonomies even more valuable for next generation retrieval solutions.

This chapter is intended to familiarize the reader with the concept of tagging and folksonomy services in general. It first introduces tagging as a new form of content categorization, distinguishing it from other categorization techniques, such as taxonomies or controlled vocabularies. Section 1.2 is dedicated to tagging in social environments and juxtaposes the two most common types of folksonomies: broad and narrow folksonomies. This is followed by a short overview of the potentials of folksonomies for next generation information retrieval solutions in Section 1.3. The final sections

[3]There exist various software tools that support users in tagging local documents, see e.g. http://www.tag2find.com or http://lunarfrog.com/taggedfrog.

[4]e.g. http://www.mendeley.com

[5]e.g. http://mail.google.com

of this chapter, Sections 1.4 and 1.5, then develop graph- and tensor-based models of folksonomy networks and describe the folksonomy datasets that will be used throughout this dissertation.

1.1 The power of tags

> "*The only group that can categorize everything is everybody.*"[6]

The shift from provider-generated to user-generated content, which has been one of the core drivers of the phenomenon commonly referred to as Web 2.0, demanded new content categorization techniques that were able to cope with open and dynamically changing environments, and could be handled by non-experts without much effort. This was the advent of tagging, and people started to assign free keywords in order to organize their blog entries, photos, bookmarks, or documents. Since then, tagging has become the predominant form of user-driven content categorization on the web, replacing or complementing previously wide-spread but rather static concepts, such as taxonomies or predefined category vocabularies. The most important benefits of tagging compared to previous categorization techniques include [Qui05]:

Inclusiveness. The set of potential tags is not limited but includes every user's views, preferences, or language as well as all potential topics.

Personalization. Users can choose the labels that match their real needs, tastes, or languages.

Low cognitive costs. Adding freely chosen keywords requires less effort than selecting categories from a potentially large set of predefined choices.

Scalability. Predefined vocabularies will start to become imprecise when a domain grows. Instead, tags can reach a nearly unlimited granularity.

Serendipity. Controlled vocabularies are designed to ease retrieval, but trade simplicity for granularity. This makes it hard to find less popular content that resides in the so-called long-tail of the information space. Tags enable users to discover long-tail information by browsing through the folksonomy network of items, tags, and users.

[6] Clay Shirky, [Shi05]

Crowd sourcing. Tagging reduces the costs of service providers, since it omits the expensive creation and maintenance of controlled vocabularies by experts.

There exist no control mechanisms in tagging. Service providers may try to influence tag assignments by recommendations, and some authors conjecture that users will tend to imitate other users' tagging behavior, resulting in de facto tag standards [Mat04]. However, our own evaluations show that, though all these tendencies exist, they do not lead to a community-wide consensus. Instead, for all analyzed broad folksonomies we generally find that users develop their personal tagging strategies without consideration of global effects. Stability only arises in situations where many users tag the same items, such that accumulative effects reduce user-level discrepancies. This phenomenon is strongly related to the *wisdom-of-crowds* metaphor [Sur04], which, applied to tagging, expects that the labeling of resources by many users will produce better descriptors than labels agreed upon by a small group of experts. The openness of tagging and the absence of control have also raised criticism. Popular arguments against tagging are [Mat04, Kro05, Qui05]:

Lack of precision. The variability of language, personal tastes, or missing expertise is expected to result in less precise tags.

Lack of recall. Users will not produce labels for all aspects of the labeled content. Even though tags go beyond binary classification schemes, such as taxonomies, they will not reach the same content coverage as indexing approaches.

Lack of hierarchy. Tags are not hierarchic. This makes it harder to find information in a top-down manner. However, existing studies show that tag hierarchies can be derived from tag co-occurrence patterns [CBSL07, LEC07].

Synonyms & Ambiguity. Users will assign different tags for the same concepts. Other tags may refer to more than one concept, lowering their value for information retrieval. Synonyms and ambiguity are well-known problems in information retrieval, and various publications have shown that the meaning of a tag can be estimated by looking at different forms of context, e.g. its co-occurrence with other tags [AYGS09, CBHS08].

Personal perspective. Users will assign the tags they consider most appropriate and will have limited interests in a global tag convergence.

This reduces the value of tags for other users. Nevertheless, consistent with the studies by other authors, our own evaluations show that the collaborative tagging of resources in broad folksonomies leads to the emergence of descriptive and stable tag distributions due to accumulative effects [GH06]. In Chapter 8 we will introduce a novel approach that infers the meaning of tags based on who assigns them. This will enable us to translate a user's individual tags to a global vocabulary, and vice versa.

Low findability. Most arguments against tagging listed above will lower the efficiency of direct tag-based retrieval solutions. These effects can be neglected for popular items, since accumulative effects reduce the variance of user-level differences. As we will show throughout this dissertation, tag clustering (Chapter 7) or tag translation (Chapter 8) may further help to increase findability.

Re-tagging. Tags reflect a user's perspective of an item at a certain point of time. Shifting user interests, changing tagging strategies, or forgotten labels thus require a continuous re-tagging of content collections.

This criticism is justified. However, for many services there currently exist no real alternatives to tagging [Qui05]. Instead, researchers have tried to increase the findability of items by developing better tag models, deriving hierarchies from tag patterns, or by mapping personal tags to community tags, as will be described in Chapter 8. Tagging can be a helpful tool for the individual organization of document repositories, emails, files, or other resources. However, of special interest from an information retrieval point of view are situations where tagging happens in a social environment.

1.2 Broad and narrow folksonomies

Services that center around the tagging activities of their users have been coined "folksonomies", an acknowledgment to the terms "folk" and "taxonomy". This term was introduced by Thomas Vander Wal in early 2004 [Wal04] shortly after the launch of now famous folksonomies like Flickr and Delicious. Vander Wal describes folksonomies as

> "*... the result of personal free tagging of information and objects ... for one's own retrieval ... in a social environment (usually shared and open to others)*".[7]

[7] Thomas Vander Wal, [Wal04]

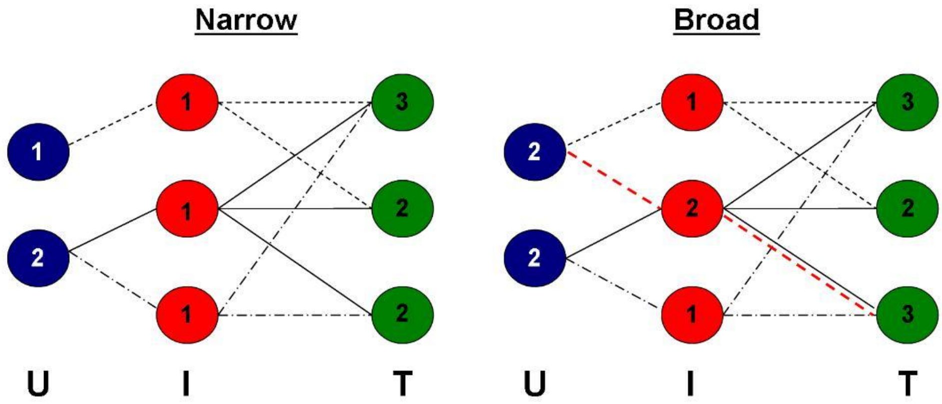


Figure 1.1: The structural difference between narrow and broad folksonomies. Blue, red, and green nodes refer to users, items, and tags respectively. Lines of the same style belong to the same post/bookmark. Numbers denote the number of posts a node occurs in. Items in narrow folksonomies can only be posted/tagged by a single user, generally the owner of the item. They therefore appear in only one post. Broad folksonomies, on the other hand, allow the free posting/tagging of items, resulting in heterogeneous item popularities and non-uniform tag distributions.

Folksonomies are traditionally divided into two types: narrow and broad folksonomies [Wal05]. This distinction originates from different posting rights. Items in narrow folksonomies can be posted, and thus tagged, by a limited number of users only. One classical example of a narrow folksonomy is Flickr, where photos are posted and labeled by their owner. Blogs that support tagging are another common example of narrow folksonomies, since the author of a blog entry will generally be the only user who assigns labels. On the other hand, broad folksonomies do not restrict posting rights, but allow every user to post and label the items she wants. Prominent examples of broad folksonomies are social bookmarking services, such as Delicious, CiteULike, or Last.fm[8].

The distinct tagging rights lead to very different structural patterns. Items in a narrow folksonomy like Flickr occur in only one post, whereas items of a broad folksonomies can appear in as many posts as there are users. The popularity of an item in a broad folksonomy can hence be estimated from the number of bookmarks that refer to it, which is not possible

[8] http://www.last.fm

in narrow folksonomies[9]. This difference is also reflected in the tags that describe resources. Items of narrow folksonomies feature flat tag distributions, since every tag is assigned exactly once. All tags thus have the same a priori weight for an item, and the ranking of tags, i.e. the order of their assignment, was shown to provide little evidence about gradations [WYYH09]. On the other hand, the aggregation of tags from different users in broad folksonomies generates non-uniform tag distributions. This makes it possible to estimate the relevance of a tag from the number of times it was assigned. Previous work further showed the emergence of characteristic tag distributions for resources in collaborative tagging services [GH06]. These distributions tend to exhibit power-law characteristics, with very few tags occurring very frequently and the majority of tags being rarely assigned. This agreement on descriptive item labels by many users is expected to boost the semantic quality of labels [Sur04]. The differences between narrow and broad folksonomies from a graph perspective are depicted in Figure 1.1. The figure also illustrates that narrow folksonomies are a special case of broad folksonomies with the constraint that each item links to exactly one user.

The fact that item distributions in broad folksonomies follow power-laws implies that a high percentage of all items will only occur in one bookmark (see Chapter 2). The amount of information about these items in the long-tail is equivalent to the situation in narrow folksonomies. The distinction between narrow and broad folksonomies is thus less strict from a data analysis perspective than it seems at a first glance.

This dissertation only investigates broad folksonomies, and most of the proposed methods rely on collaborative tag assignments, which impedes their direct deployment in narrow folksonomies. However, this dissertation will also consider insights from the research on narrow folksonomies if these are not bound to the folksonomy type. For the rest of this work we will use the term "folksonomy" in the sense of "broad folksonomy" if not stated otherwise.

1.3 Folksonomies & IR

Aggregating the interests of up to millions of users, folksonomies reflect the dynamics of the underlying domains. High traffic websites will likely reappear among the popular resources in a community like Delicious [HKGM08].

[9]This does not mean that there is no way to fame in narrow folksonomies. Videos on YouTube gain popularity the more people watch them. Photos on Flickr can be ranked by the number of users that have added them as their *favorites*. Also, many narrow folksonomy services support the commenting of other users' content. The number of comments will likely correlate to the popularity of an item.

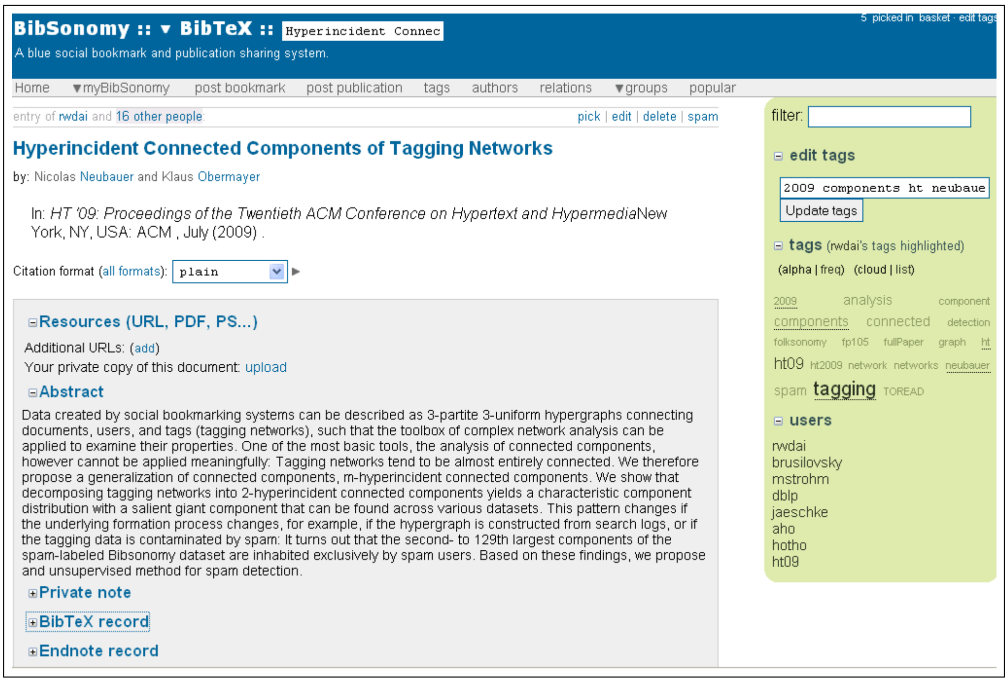


Figure 1.2: Interface of the Bibsonomy social bookmarking service. The image shows bibliographic details for a selected publication. A navigation component on the right side displays the assigned tags in form of a tag cloud as well as the users who bookmarked the selected paper. Clicking on one of the tags or users in this panel opens a dedicated website. Users that want to discover new content can thus browse the folksonomy graph along its edges.

Furthermore, external trends, e.g. the advent of a new website or a new influential blog entry, were found to reappear in folksonomies almost without any delay [HKGM08]. Combining these characteristics with the existence of user-generated annotations, folksonomies have become an invaluable source for information retrieval (IR). These benefits come at a low cost, since all information is centrally available, and no continuous, distributed crawling process as for web indexing is required.

All folksonomies mentioned in this dissertation offer tag-based search functionality that works very similarly to well-known search engines. Upon entering a set of query tags, the user is presented a result list of the most recent postings that have been labeled with the query tags, or a list of items that have a high general co-occurrence with those tags. However, the search functionality of common folksonomy services is rather primitive and lacks more advanced features, e.g. the stemming of tags, i.e. the reduction of tag terms to their roots, personalization, or latent indexing. A user that searches for relevant content thus has to guess the tags that other users considered relevant.

However, information retrieval in folksonomies goes well beyond tag-based search scenarios. Additional ways of finding information come from the interface design of most folksonomies. Figure 1.2 displays the interface of the Bibsonomy social bookmarking service. The image shows the bibliographic details for a selected publication. The green component on the right lists the tags that have been assigned to the publication as well as the users that bookmarked it. Clicking on one of the tags will open a view with items it was assigned to. Selecting a user instead opens a dedicated website that lists this user's bookmarks, publications, and tags. This interface design permits the discovery of interesting items, users, or topics by browsing the folksonomy graph along its edges. Similar interfaces can be found for all the folksonomies discussed throughout this work.

One visualization technique that plays an important role in this discovery process are *tag clouds*. Figure 1.3 displays Flickr's "explore by tag" view that consists of three distinct tag clouds: the most popular global tags, the most popular tags in the last week, as well as the popular tags in the last 24 hours. The popularity of a tag within the cloud is accentuated by dynamic font sizes. These tag clouds act as global entry points to the folksonomy by emphasizing the topics that are (currently) popular within the community. Selecting a tag will show the items that have been recently labeled with it or items of high relevance. Tag clouds can also visualize the tags that are related to the current context, e.g. the tags commonly assigned to the current item. Thus, they provide a quick overview of the topics that characterize an object or are related to it, facilitating the discovery process.

Data from folksonomies can also help improve existing search solutions. One popular scenario is the exploitation of tags for better item and user models. The authors of [RHMGM09] propose the incorporation of tags of Delicious users to derive better models of websites, with the goal of improving the accuracy of existing search engines. This approach is especially appealing, since it allows to model all types of potentially heterogeneous website content in a unified way. Other previous research utilized the data of folksonomies for search personalization. For instance, Noll and Meinel [NM07] derive user preferences from a user's tagging activity in Delicious. The authors then utilize this additional knowledge to re-rank the results of common search engines with respect to these preferences. We will extensively discuss folksonomies from an information retrieval perspective in Part II of this dissertation.

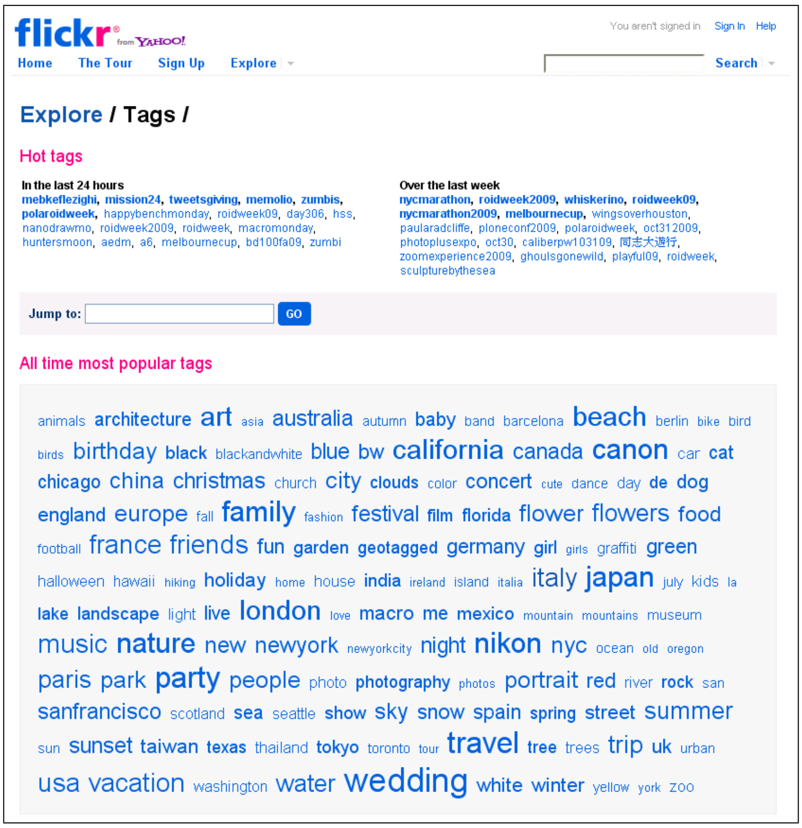


Figure 1.3: "Explore by tag" interface of Flickr showing the most popular community tags as tag clouds. Popularity is emphasized by dynamic font sizes. Selecting a tag lists the most recent photos this tag was assigned to. Tag clouds are part of all major folksonomy services.

1.4 Modeling folksonomies

Notation

In accordance with common notation practice, throughout this work bold calligraphic letters will be used for tensors ($\boldsymbol{\mathcal{Y}}$), bold capital letters for matrices ($\mathbf{A}$), and bold lower case letters for vectors ($\mathbf{u}$). Sets and set elements are represented by non-bold letters, e.g. I and i. Capital letter subscripts of tensors and matrices denote the sets corresponding to each mode, such that the matrix $\mathbf{A}_{IT}$ represents a $|I|$-by-$|T|$ matrix with rows and columns built from the sets I and T respectively. Lower case letter subscripts represent the reduction to only one entry of the corresponding set, e.g. $\mathbf{A}_{it}$ is the entry of $\mathbf{A}_{IT}$ that resides in row i and column t. Analogously, we use subscripts for sets to indicate the members of the original set that fall into the context represented by the subscript, e.g. I_u represents all items a user u has bookmarked.

Model

According to [HJSS06a], a folksonomy can be described as a tuple $F := (I, T, U, Y)$ where I, T, U are finite sets of items, tags, and users:

$$\begin{aligned} I &= \{i_1, \ldots, i_k\} \\ T &= \{t_1, \ldots, t_l\} \\ U &= \{u_1, \ldots, u_m\}. \end{aligned}$$

Y is a ternary relation whose elements are called *tag assignments*. A *tag assignments* (TAS) is defined by the authors as a relation $Y \subseteq U \times T \times I$, so that the tripartite folksonomy hypergraph is given by $G = (V, E)$, where $V = I \cup T \cup U$ and $E = \{(i, t, u) | (i, t, u) \in Y\}$. Bookmarks can then be formalized as the sets of tag assignments that share the same user-item entities $B = \{(i, u) | \exists t : (i, t, u) \in Y\}$. However, this formalization disregards bookmarks without any assigned tags. In order to include these bookmarks, we additionally introduce a virtual node of type T called the "no_tag" tag. The tag vocabulary of a user, her *personomy* [HJSS06a], is given as $T_u = \{t | \exists i : (i, t, u) \in Y\}$, the tag vocabulary of an item as $T_i = \{t | \exists u : (i, t, u) \in Y\}$, and the tags of a bookmark as $T_{iu} = \{t | (i, t, u) \in Y\}$. This tripartite hypergraph structure is characteristic for all broad folksonomies.

A natural way to represent the ternary relations Y that build a folksonomy are tensors, as shown in Figure 1.4. Tensors, as used in this dissertation, are multi-dimensional matrices that go beyond the common 2-dimensional row-column representation. Folksonomies instead require a 3-mode model that captures the incidence of items, tags, and users. We construct the 3-mode folksonomy tensor $\boldsymbol{\mathcal{Y}}_{ITU} \in \mathbb{R}^{|I| \times |T| \times |U|}$ that represents Y as

$$\boldsymbol{\mathcal{Y}}_{itu} = \begin{cases} 1, & \text{if } (i, t, u) \in Y \\ 0, & \text{otherwise.} \end{cases}$$

Modeling folksonomies as tensors allows us to apply tensor calculus in order to determine substructures of Y. For example, the co-occurrence matrix $\mathbf{A}_{IT}$ of items and tags can be calculated by aggregating the tags assigned to each item over all users. This is equivalent to the calculation of

$$\mathbf{A}_{IT} = \boldsymbol{\mathcal{Y}}_{ITU} \,\overline{\times}_U\, \mathbf{u}^1, \tag{1.1}$$

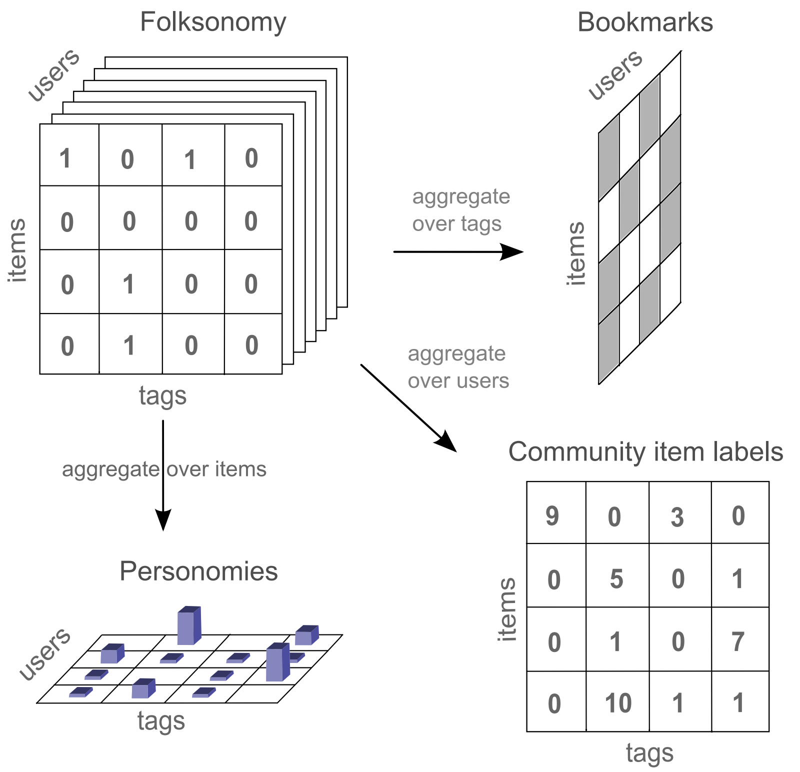


Figure 1.4: A folksonomy as a 3-mode tensor, where modes correspond to items, tags, and users respectively. The figure further shows the resulting matrices when the tensor is collapsed in one of its modes.

where $\overline{\times}_X$ is the vector product with a tensor in the mode represented by X, and $\mathbf{u}^1$ is a vector of size $|U|$ of all ones. It is likewise possible to aggregate along other modes as shown in Figure 1.4. Using the vector product we can also derive the tags a user assigned to an item as

$$\mathbf{t}(i, u) = \boldsymbol{\mathcal{Y}}_{ITU} \,\overline{\times}_I\, \mathbf{i} \,\overline{\times}_U\, \mathbf{u}, \tag{1.2}$$

where $\mathbf{i}$ and $\mathbf{u}$ are vectors of all zeros except for the indices that correspond to the user-item pair and which are set to 1. Notation-wise, this is equivalent to $\boldsymbol{\mathcal{Y}}_{iTu}$, which in turn is the vector representation of the set T_{iu}.

1.5 Datasets

The analyses and evaluations in this work are conducted on a selection of real-world folksonomy datasets, which we introduce in this section.

Delicious. The Delicious bookmarking service is probably the best researched folksonomy to date. It allows its users to centrally save and organize their bookmark collections on the web. To conduct our analyses on a large-scale snapshot of Delicious, we initiated a crawling process that constantly downloaded objects from the Delicious website. This process ran from December 2007 to April 2008 and produced a dataset of 132.5 million bookmarks and 420 million tag assignments from 948 thousand users. This makes it by far the largest snapshot of a real-world folksonomy ever analyzed in academia. The full corpus has been thoroughly described in [WZB08] and [WZB10a]. Furthermore, subsets of this dataset have been used for evaluations in [WUS09], [WPK+08], and [WZB10b]. The full dataset was made publicly available for other researchers[10]. We will present an extensive analysis of this dataset (*delicious*) in Chapter 2.

A vast amount of the originally crawled Delicious bookmarks originates from spam activity, which impedes meaningful analyses and evaluations (see Chapter 4 for details). We therefore applied spam filtering to the originally retrieved *delicious* dataset, as described in Section 4.2. We will refer to the resulting spam-filtered dataset as $delicious_{clean}$.

CiteULike. The CiteULike service supports researchers who want to collect, manage, and share publications on-line. Users are asked to provide missing bibliographic information as well as representative tags during the bookmarking process. CiteULike is the only among the examined services that provides daily snapshots of the full community for download. This undemanding data accessibility makes CiteULike an attractive research object, e.g. [SWUH09], [OGC08], and [ZC08]. The CiteULike snapshot considered here (*citeulike*) was downloaded on March 23rd, 2009 and consists of about 1.6 million bookmarks that link to 1.4 million different publications.

Bibsonomy. The Bibsonomy social bookmarking service[11] is another highly researched folksonomy that lets users collect and annotate URLs as well as publications. This work considers two different publicly available Bibsonomy datasets. The $bibsonomy_{spam}$ dataset was released as part of the 2008 ECML PKDD spam detection challenge[12]. It consists of more than 2.2 million Bibsonomy bookmarks. However, about 92 percent of these bookmarks have been manually classified as spam.

[10] http://www.dai-labor.de/IRML/datasets
[11] http://www.bibsonomy.org
[12] http://www.kde.cs.uni-kassel.de/ws/rsdc08/

We will use this dataset for the evaluation of our spam detectors in Section 4.2. All other analyses as well as the evaluations of the developed recommender systems are performed on the larger and spam-free $bibsonomy_{2009}$ corpus instead. This corpus has been released as part of the 2009 ECML PKDD discovery challenge on tag recommendation[13].

Both Bibsonomy datasets contain all bookmarks that were publicly available by the time of dataset creation. We convert the Bibsonomy network structure to a standard tripartite folksonomy graph by merging URLs and publications into a single item set, as previously done by other authors [JMH+07, HJSS06a, SNM08, RBMNST09].

MovieLens10M. The MovieLens film rating datasets[14] have become popular benchmark corpora for recommendation engines. Each of the different dataset versions consists of the ratings that users have given to the movies offered by the service. Ratings range from one to five with five being the best. All users within the datasets were selected at random and have rated at least 20 movies. Furthermore, the most recent version of the dataset (*movielens10M*) also includes the tags users have assigned to movies. Previous work has shown that these tags can help improve user models, resulting in better movie recommendations [SVR09]. However, since tagging is a rather new feature of MovieLens, most movies remain untagged. Nevertheless, we consider MovieLens in our analyses, since it allows us to compare our findings in other folksonomies to a well known and established recommendation dataset.

The *movielens10M* dataset differs in nature, since users express their likes and dislikes by ratings. To make the dataset comparable, we discard all rating values and consider a user as connected to a movie if she has rated it. This simplification contains the information structure between movies, tags, and users. The loss of the actual rating information can be neglected here, since we limit our investigations of this dataset to structural characteristics. For other methods of converting rating datasets to a link-based representation see [DDGR07].

From all datasets, we only consider the network structure between items, tags, and users and discard all additional information, such as bibliographic data in the case of publications or user-provided descriptions of URLs. The resulting global network statistics for all datasets are displayed in Table 1.1.

[13] http://www.kde.cs.uni-kassel.de/ws/dc09

[14] http://www.grouplens.org/node/73

Table 1.1: Global statistics of all folksonomy datasets.

	$\lvert E\rvert$	$\lvert B\rvert$	$\lvert I\rvert$	$\lvert T\rvert$	$\lvert U\rvert$
delicious	420,026,460	132,500,391	50,222,260	6,204,149	947,835
$delicious_{clean}$	308,207,242	108,777,128	36,481,193	4,535,198	854,209
$bibsonomy_{spam}$	16,818,699	2,241,824	1,574,983	396,474	38,920
$bibsonomy_{2009}$	1,401,104	421,928	378,378	93,756	3,617
citeulike	5,215,824	1,662,203	1,433,585	294,213	44,064
movielens10M	10,020,295	10,000,051	10,676	9,530	69,878

In the next chapter of this work, we will perform an extensive analysis of the bookmarking and tagging patterns in folksonomies on the example of the Delicious social bookmarking service. We choose Delicious, since it is the most representative dataset and its size permits investigations that cannot be performed on smaller corpora. We will extend our analysis to other folksonomies when discussing the general network properties of folksonomies in Chapter 3.

Chapter 2

A delicious endeavor

Delicious is one of the most popular and best researched social bookmarking communities to date. The service allows its users to centrally collect and share their bookmarks. These bookmarks can refer to web resources of any type as long as these resources can be identified by a URL. Since Delicious is a web application, it can be accessed from everywhere and independently of the current device. Delicious was also one of the first services that relied solely on tags for content categorization, offering its users the chance to organize their collections of bookmarks by freely chosen keywords. The organizational aspect is complemented by community features, and users can add other users to their network in order to forward bookmarks[1]. Furthermore, by browsing the bookmark collections of other users they can see what their friends or other members of Delicious considered worth bookmarking.

When posting a bookmark, users can provide a description, by default the title of the website, an extended description, and a list of tags. In order to simplify the bookmarking process and to support convergence, Delicious suggests tags that were frequently assigned to the bookmarked resource. Users can additionally mark a bookmark as *private* so that it remains hidden to the community. A bookmark can be added by going to the Delicious website or, more conveniently, by using one of the many browser plug-ins that make bookmarking with a social bookmarking system and later retrieval as easy as browser-based bookmarking. Many websites, first and foremost blogs and news sites, also provide direct links for bookmarking their content to various bookmarking services including Delicious.

The channels of information diffusion in Delicious are manifold. Using an RSS[2] service, users can subscribe to other users' bookmarks or to topics

[1] Since August 2009 Delicious further supports the sending of bookmarks via email or sharing them on Twitter [Del07b].

[2] Really Simple Syndication

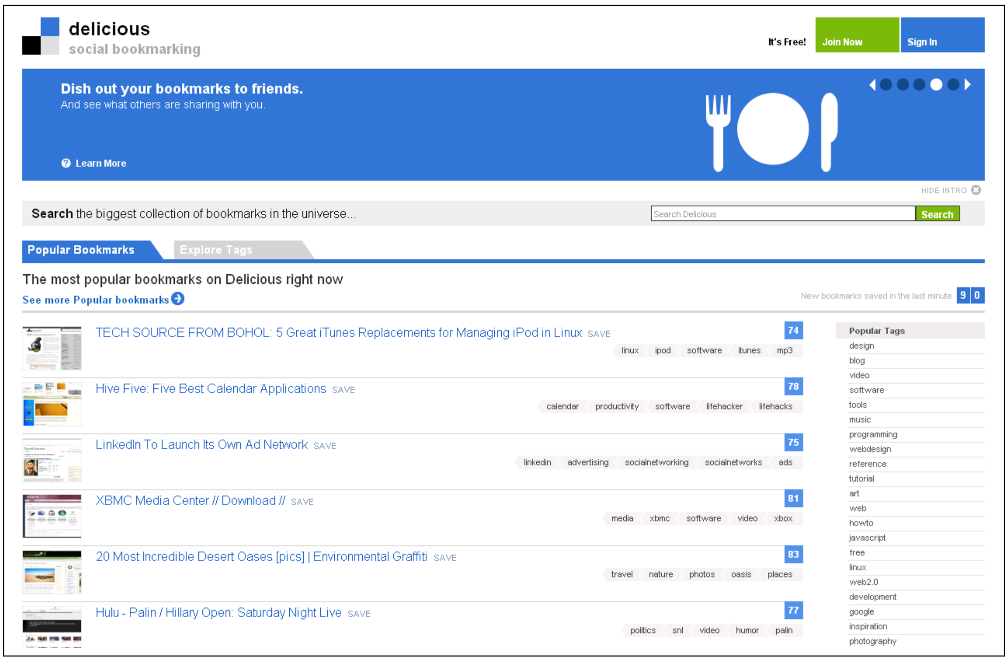


Figure 2.1: The Delicious main page showing the currently most popular web resources and tags (September 2008).

represented by tags. This way, users receive updates in real-time about developments in their friends or colleagues interests or about new URLs appearing in a certain domain. Another channel is the Delicious main page as shown in Figure 2.1. Once a resource reaches a certain popularity, it appears on this page and is therefore likely to attract the attention of other users. It is equally possible that a bookmark reaches other users via the Delicious feed of recently added bookmarks. Finally, users can forward bookmarks within their network of friends, colleagues, or other members of interest, which emphasizes the social aspect of bookmarking services.

Delicious also offers various ways for finding information. The most obvious way is the Delicious search interface that is shown in the upper right corner of Figure 2.1. This interface supports text-based queries which are matched against tags as well as URLs. Furthermore, similar to Bibsonomy, the Delicious website allows to browse the folksonomy graph along its edges. For instance, starting from the most popular page of Figure 2.1, a user can discover other content by following links to related tags or users. The quality of content can be estimated from the number of users that connect to it, shown by the small blue boxes in Figure 2.1. Furthermore, a user can discover interesting content within the collections of her friends or other members of her network.

Delicious went on-line in September 2003 [Del07a], and its community of users and their corresponding bookmark collections has been growing ever since. According to our dataset, there were at least $7,305,559$ newly added bookmarks and $47,429$ new users in December 2007. The collective activity of all Delicious users resulted in more than 55 million collected resources, with the majority of them categorized by tags.

2.1 Crawling Delicious

There exists no perfect strategy for crawling Delicious or other folksonomies. One of the most straightforward approaches, chosen for example by [GH06] or [HKGM08], is the monitoring of the Delicious pages that list the most recent bookmarks or currently popular URLs. Despite its simplicity, this crawling method is rather time consuming and does not allow for crawling old posts. Furthermore, we find that the recent feed is not complete and only contains a small percentage of the entire posted content. What kind of filtering criterion has been applied remains unclear. Crawling the recent feed therefore hardly results in representative data sets. However, it may produce a set of graph nodes that act as seed for further crawling.

Another crawling approach follows the links of the user to user network [ALMP08]. This approach is expected to limit the percentage of spam in the resulting data, since spammers are unlikely to be friends with other users. Despite this advantage, the method will produce a dataset biased towards inter-connected users, whereas users without connections within the friendship graph will not appear. Furthermore, we did not want to discard spam users, since we were particularly interested in their impact on the service.

The most common method to crawl Delicious or similar services is to follow the links of the tripartite hypergraph between users, tags, and items [HKGM08, HJSS06a, Mik05]. As a starting point we crawled all bookmarks related to the tag "web2.0". From the resulting set of bookmarks we extracted all related tags and recursively used these for further queries. As a result of this process, we retrieved 45 million unique bookmarks.

During our retrieval process, we found that the Delicious service limits the number of returned bookmarks when queried tag-wise. This led to an incomplete representation of frequent tags within the initial corpus. The limitation in the number of query results seems to occur only for the tag-wise retrieval, while user-wise or item-wise queries are not affected. We therefore downloaded all public bookmarks of the most active users within the initial corpus. The crawled dataset consisted of 142 million Delicious bookmarks from 978,979 users between December 2007 and April 2008. In parallel to the main crawling process, we monitored the Delicious page of recent bookmarks. We found that around 85 percent of the bookmarks listed on this page had also been crawled by the main process. We therefore conclude that the dataset represents a large fraction of the entire Delicious service.

To balance the corpus, we removed all bookmarks posted before January 1st, 2008 and only kept URLs starting with "http://". This reduced the initial corpus to 132.5 million bookmarks and 420 million tag assignments from 947,835 users. The full *delicious* dataset is made publicly available for other researchers upon request[3].

The result of a random crawler that follows hypergraph links tends to be biased towards frequent items. Furthermore, as our initial, tag-wise crawl was biased towards recent items, we expect this bias to reappear in the *delicious* corpus. Finally, since we only crawled the most active users from the initial data set, users with a low participation rate are under-represented. We will consider these limitations of the *delicious* dataset when discussing our findings throughout the next sections of this thesis.

2.2 Bookmarking patterns

In this section we try to answer various questions in order to achieve a better understanding of the actual bookmarking behavior of the Delicious community. Understanding the bookmarking patterns within Delicious and other folksonomies will later help us in the design of more accurate recommenders. This section focuses on the interaction between users and items within Delicious, whereas tagging patterns will be discussed in Section 2.3.

What do users bookmark?

Table 2.1 lists the most popular URLs in our corpus. The list is thematically split into websites related to social resource or knowledge sharing (entries 1, 2, 5, 6, 7 and 8) as well as web development (entries 4, 9 and 10). Not surprisingly, the Delicious community is biased towards content related to web communities and (web) technology. However, most web portals use a deep-link structure for content access, such that the aggregation over URLs does not provide the full picture. To credit deep-links, we aggregate all URLs to domains. The 10 domains that have been bookmarked most often in the *delicious* dataset are listed in Table 2.2. Here, we find news providers and

[3] http://www.dai-labor.de/IRML/datasets

Table 2.1: Top 10 most frequent URLs in the *delicious* corpus.

	URL	**bookmarks**
1.	http://www.flickr.com	33,222
2.	http://www.pandora.com	32,634
3.	http://www.netvibes.com	26,743
4.	http://script.aculo.us	26,082
5.	http://slashdot.org	25,272
6.	http://en.wikipedia.org/wiki/Main_Page	23,983
7.	http://www.youtube.com	23,530
8.	http://www.last.fm	22,757
9.	http://oswd.org	20,430
10.	http://www.alvit.de/handbook	20,230

Table 2.2: Top 10 most frequent domains in the *delicious* corpus.

	domain	**bookmarks**	**users**
1.	http://en.wikipedia.org	919,465	205,639
2.	http://www.youtube.com	915,789	186,326
3.	http://www.flickr.com	535,176	162,363
4.	http://www.nytimes.com	503,776	101,375
5.	http://www.google.com	392,360	156,990
6.	http://lifehacker.com	368,078	90,628
7.	http://www.amazon.com	341,414	91,073
8.	http://news.bbc.co.uk	317,978	75,610
9.	http://www.microsoft.com	290,501	101,947
10.	http://community.livejournal.com	280,020	29,655

enterprise portals among the Top 10 items, i.e. services that host articles or other content in the form of deep-links.

Do users stick to the same information sources?

A Delicious user can bookmark a URL only once, but she can bookmark as many links from the same domain as she likes. As a result, we find that the number of bookmarks and the number of users per domain tend to differ. For instance, Table 2.2 underlines the fact that enterprise domains tend to have a lower bookmarks per user rate than domains with highly dynamic content, such as news sites where users follow the news feeds and bookmark articles of interest to them. Users thus tend to return to information sources and to bookmark diverse content from the same providers. The most loyal audience of all domains listed in Table 2.2 belongs to the community of the

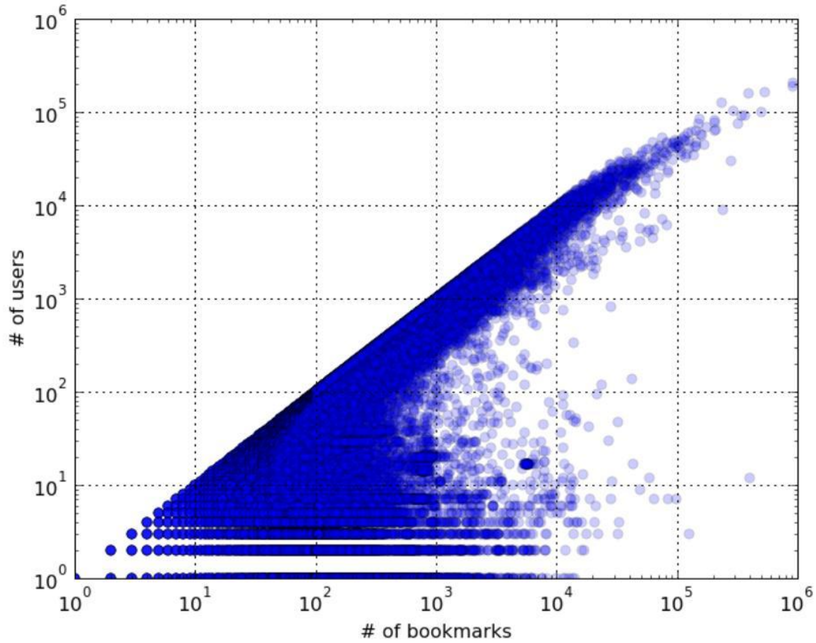


Figure 2.2: The number of bookmarks compared to the number of users linking to the same domain in Delicious (*log-log scale*). Many domains, such as blogs or Wikipedia, provide various content in the form of deep-links and can thus be bookmarked more than once per user. Other services without deep-link structure can be bookmarked only once per user. Domains with a very imbalanced ratio of bookmarks to users often originate from spammers that try to promote their domains with many posts (see Section 4).

LiveJournal web portal. Here, every interested user on average contributes around 10 bookmarks, which is twice the rate of Wikipedia. Because of this potential user concentration, we find that the popularity of a domain within Delicious when measured in bookmarks does not necessarily reflect its popularity among the Delicious user base. User-oriented measures, such as the *diffusion-of-attention* concept which we will introduce in Section 4.2, may thus be better suited for the estimation of domain popularities.

The clustering of bookmarks around domains in the *delicious* corpus is also shown in Figure 2.2. The figure plots the number of users that have bookmarked a domain at least once against the number of bookmarks per domain. We find that there exist domains that are consistently bookmarked only once by users. This indicates that these domains do not exhibit a deep-link structure. On the other hand, some very frequent domains are frequented by a few users only. Most of these domains are related to spam within the Delicious dataset. These spammers try to promote their web domains by vast amounts of posts, as will be described in Section 4. However, a less drastic clustering can be observed for web portals with highly dynamic content, like news portals or blog sites.

How do user interests compare to web directories?

Web directories are often considered the predecessors of social bookmarking services [O'R05]. They aim at structuring web resources into a predefined topic taxonomy, allowing users to find interesting content by browsing the directory. Websites are reviewed by expert teams before they are added to the corresponding taxonomy branch. Web directory services thus try to structure the web in a top-down manner, contrasting the bottom-up categorization process of folksonomies. However, with the exponential growth of the web, web directories became more and more expensive to maintain, loosing ground to search engines and social bookmarking services. This paradigm change is best reflected by the product policy of Yahoo!. Yahoo! pioneered web directories on their homepage in addition to their standard search engine, spending much time and effort in manual website classifications. However, in December 2005 Yahoo! acquired Delicious [Zaw05], whereas the web directory was removed from the Yahoo! portal.

Another very prominent web directory is the Open Directory Project[4] (ODP), which can be freely downloaded[5]. To compare the bookmarking activity of Delicious users with the efforts of the ODP, we downloaded the full ODP directory and all attached URLs. This download took place in February 2008, the same time we crawled Delicious. The downloaded ODP dataset consists of around 4.4 million URLs classified into nearly 600,000 topics. Of all ODP URLs around 800,000 also appear in our *delicious* dataset. These URLs have been bookmarked more than 10 million times.

The top level of the ODP taxonomy consists of 17 categories. Figure 2.3 shows that the majority of overlapping Delicious URLs belongs to the "World" and "Regional" categories, which cover non-US websites as well as regional content. Counts are plotted on a logarithmic scale, such that differences are larger than they appear. The high overlap in these categories results from the overall bias of the Open Directory Project towards these two topics. However, the plot also emphasizes that URLs from these top-

[4] http://www.dmoz.org
[5] http://rdf.dmoz.org/

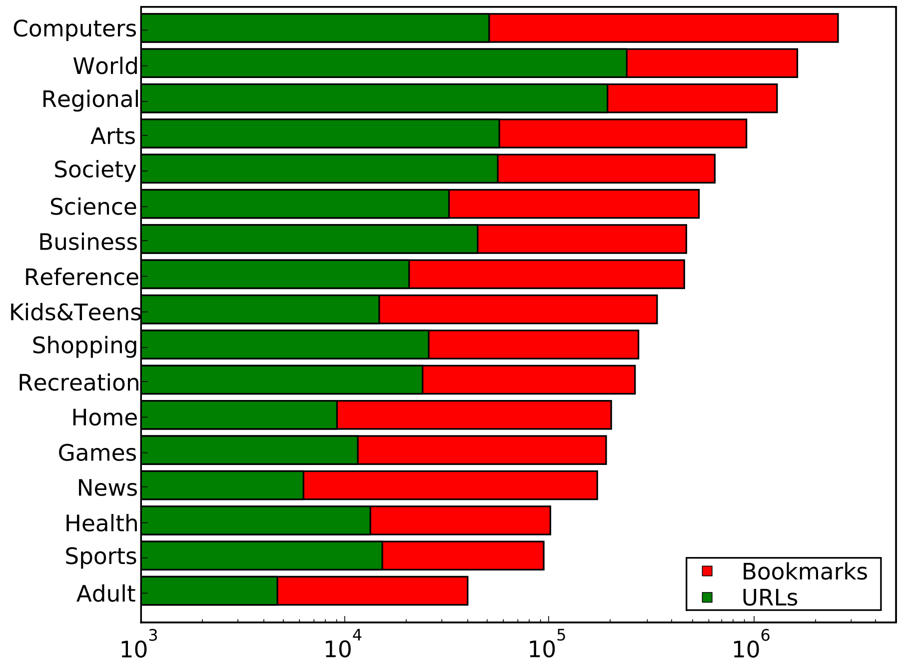


Figure 2.3: Distribution of Delicious URLs and bookmarks over Open Directory categories (log-scale). The figure supports previous findings that most bookmarks on Delicious are related to technology.

ics are less popular within Delicious. Instead we find that, although fewer URLs fall into the "Computers" category, these URLs attract more interest in the Delicious community. The overall contribution of this category to the number of bookmarks therefore exceeds the one of all other categories. This observation further emphasizes the bias of the Delicious community towards technology-related content.

When do users bookmark resources?

Golder and Huberman [GH06] observe that the Delicious community pays attention to new URLs only for a very short period of time. As a result, these URLs receive most of their posts very quickly and disappear shortly afterwards. Figure 2.4 shows the popularity of the most popular URLs in June 2006 that were unknown the month before. As can be seen, each URL peaks within a timespan of only few days before the number of posts drastically decreases. According to [GH06], this burst in popularity is likely to

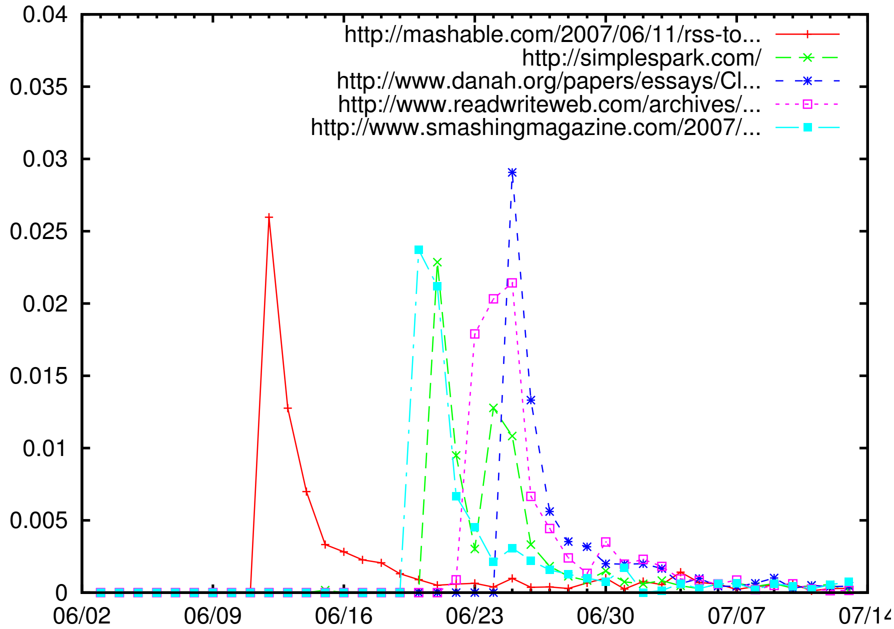


Figure 2.4: Popularity over time for the five most popular Delicious URLs in June 2006 that were unknown the month before. Popularity is given as percentage of overall bookmarks. Most upcoming URLs disappear shortly after peaking.

be caused by the appearance of a URL on the Delicious main page or by external reasons, such as the appearance of a URL on a widely read blog or in a popular Twitter feed. Another cause of an initial popularity increase could be the spread of interest within the network of Delicious users itself. This intuition is supported by recent studies of the Flickr photo sharing community, showing a strong dependency between the spread of interest and the underlying social graph [CMG09].

[HKGM08] report that URLs posted to Delicious have been often modified recently. The authors therefore argue that Delicious users tend to post active pages which are constantly updated. When comparing pages on Delicious with search engine results, they further report an actuality of posts comparable to the top results of the Yahoo! search engine. Their results also show that Delicious reacts much faster to external events than web directory services, such as the ODP. Measuring the overlap of URLs in Delicious and Yahoo!, [HKGM08] further report 25 percent of all Delicious posts to refer to non-indexed pages. These pages were not known to the search engine by

Table 2.3: Top 20 most frequent tags in the corpus

	tag	**count***		**tag**	**count***
1.	design	4.93	11.	free	2.50
2.	blog	4.02	12.	web2.0	2.42
3.	software	3.95	13.	art	2.30
4.	web	3.27	14.	linux	2.25
5.	tools	3.23	15.	css	2.21
6.	reference	3.15	16.	howto	2.17
7.	programming	3.08	17.	tutorial	1.98
8.	music	2.99	18.	news	1.96
9.	video	2.60	19.	photography	1.76
10.	webdesign	2.54	20.	business	1.71

* in millions

the time of their posting but only several days thereafter. The analysis also showed a disproportionately high coverage of URLs from the Yahoo! search results by Delicious. The authors conclude that social bookmarking services are a valuable data source for search, even though they only cover a small fraction of the overall World Wide Web. These results also highlight the potential of recommendation in such environments, a potential which is even higher when tags come into play.

2.3 Tagging patterns

This section investigates how Delicious users utilize tags in order to organize their collections of bookmarks. It further discusses global, as well as item- and user-level, patterns that emerge from this collaborative tagging activity.

Which tags are assigned?

Table 2.3 lists the 20 most frequent tags within our corpus. This list again highlights that Delicious is biased towards web and web development related topics. However, we find that the frequency of the most popular tags is some orders of magnitude above the one of items and users[6]. Furthermore, we observe that all popular tags are English words.

Golder and Huberman [GH06] report that tags can perform different functions, such as describing the type of a bookmarked resource or what it is

[6] The most active user in the $delicious_{clean}$ dataset has collected more than 24 thousand bookmarks.

Table 2.4: Tagging intentions according to [GH06]. Examples are added by the author if not given in [GH06]. Numbers in parentheses indicate the rank of a tag in the *delicious* dataset.

	Intention	**Examples**
1.	Identify what/who it is about.	design(1), software(3), web(4), tools(5)
2.	Identify what it is.	blog(2), software(3), tools(5), reference(6)
3.	Identify who owns it.	google(26), apple(61), microsoft(97), flickr(119)
4.	Refining categories.	online(40), geek(143), 2007(404), 2.0(476)
5.	Identify qualities/characteristics.	free(11), fun(37), humor(47), inspiration(48)
6.	Self reference.	self(1290), own(1334), mine(3455)
7.	Task organization.	research(53), toread(83), work(93)

about. Table 2.4 lists the functions identified by [GH06] complemented by some examples and their popularity rank within our *delicious* corpus. The authors note that the majority of tags describe the topic or nature of a resource. This observation is supported by the tags shown in Table 2.4. Of these, only the tag "free" does not fall into the aforementioned categories. The same bias was discovered in the studies by Bischoff et al. [BFNP08]. However, the latter also report that different folksonomies favor different tag types. Whereas Flickr users exhibit a strong tendency in favor of location-related tags, users of Last.fm tend to assign tags in order to express their opinion about songs.

Complementary to the observations by [GH06], we find that users often assign tags in order to organize their bookmarks, and to ease later retrieval. The importance of this retrieval aspect is also highlighted by the fact that 15 percent of all users in the data set structure their tags by aggregating them manually into *tag bundles*. *Tag bundles* are a Delicious functionality that lets users define categories (bundles) and assign the corresponding tags. *Tag bundles* then behave like (meta-)tags and can be used as filters the same way as standard tags.

Many category tags are likely to be more user specific and will hence tend to appear in the long-tail of the global tag distribution. Analyzing tag functions from a global perspective only [GH06, BFNP08] thus underestimates the function of tags as content organizers. Furthermore, the need for organization emerges once a user reaches a certain number of bookmarks, such that this intention may have been less present in the beginning of Delicious when [GH06] performed their analysis. Support to this latter assumption is provided by the observation that in the beginning of Delicious a high number of bookmarks was not tagged at all.

How many tags are assigned?

We find the average Delicious bookmark to be labeled with 3.16 tags. Similar tag rates can also be found for the *bibsonomy*$_{2009}$ (3.3) and *citeulike* (3.13) datasets. However, for Delicious this number is not constant over time and strongly varies among users. Figure 2.5 shows a rise in the number of tags per bookmark from the start of Delicious till December 2007. This effect is partly caused by an increase of spam users who assign a large number of tags to make their posts visible. In fact, for the $delicious_{clean}$ dataset the average number of tags results significantly lower (2.77). However, we also observe a constantly increasing tagging rate for non-spam users, saturating in the end of 2005 at around 2.8 tags per bookmark. We believe this can be explained by constantly growing bookmark collections and the resulting need for more tags in order to categorize and retrieve content. In the beginning of the service, users could distinguish their bookmarks using only a limited number of common tags. However, with growing bookmark collections more tags were needed for structure and transparency. Furthermore, we assume that in 2003, when Delicious went on-line, most users were not yet familiar with tagging at all. Thus, the increase in tag assignments over time in the initial phase of Delicious also reflects the adaption of the concept of tagging itself. This fact is further emphasized by Figure 2.6 which plots the ratio of bookmarks without any labels over time. Here, we initially find a constant decrease in the percentage of unlabeled bookmarks. Whereas in the beginning of Delicious approximately 18 percent of all bookmarks were untagged, this number changed to approximately 5 percent only one year later when tagging became popular with the advent of Web 2.0 services.

The rise in untagged posts in recent years is mainly caused by spammers who post large numbers of untagged content. This activity culminates in December 2007 when the bulk upload of untagged posts by a single spammer causes an increase in the portion of non-tagged elements by 10 percent. However, we also encounter an increase of untagged resources for the spam filtered $delicious_{clean}$ dataset, which could indicate imperfect spam filtering.

The number of tags per bookmark also strongly depends on the user. Figure 2.7 plots the number of average tag assignments per user compared to the size of this user's bookmark collection. It can be seen that some extreme “users” assign more than one hundred tags on average. This further highlights the impact of spam.

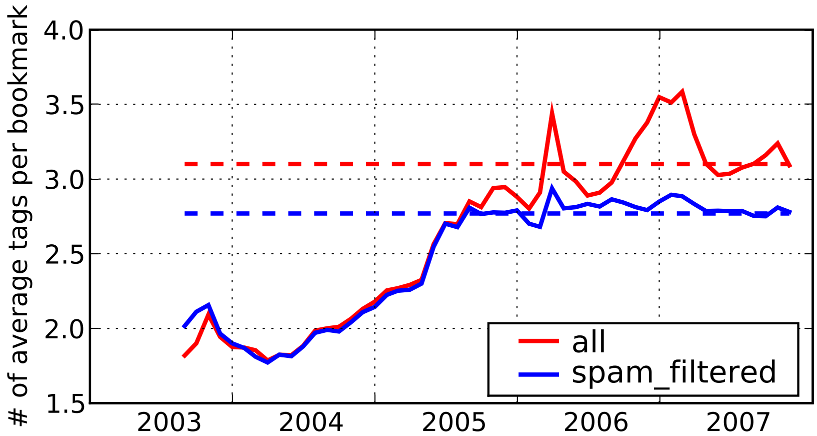


Figure 2.5: Average number of tags assigned to a bookmark over time for the complete *delicious* and the spam-filtered $delicious_{clean}$ datasets (see Section 4.3). The dotted lines represent the overall average when months are weighted by the number of bookmarks. The plot shows the adaptation of the tagging concept by Delicious users over time till saturation. Spammers generally tend to assign more tags to increase the visibility of their sites.

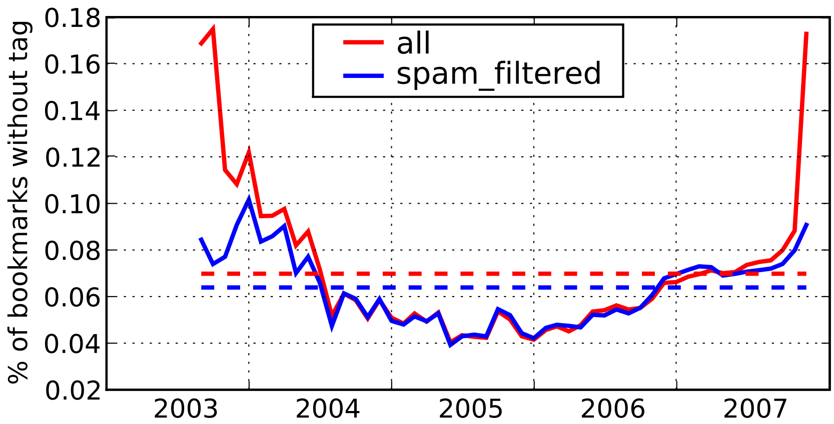


Figure 2.6: Percentage of bookmarks without any tag assignment for the complete *delicious* and the spam-filtered $delicious_{clean}$ datasets (see Section 4.3). The dotted lines represent the overall average when months are weighted by the number of bookmarks. The number of posts without any tag assignment decreases over time. It can also be seen that some spammers do not care about tagging their bookmarks at all. The increase of untagged bookmarks in recent years could result from imperfect spam filtering.

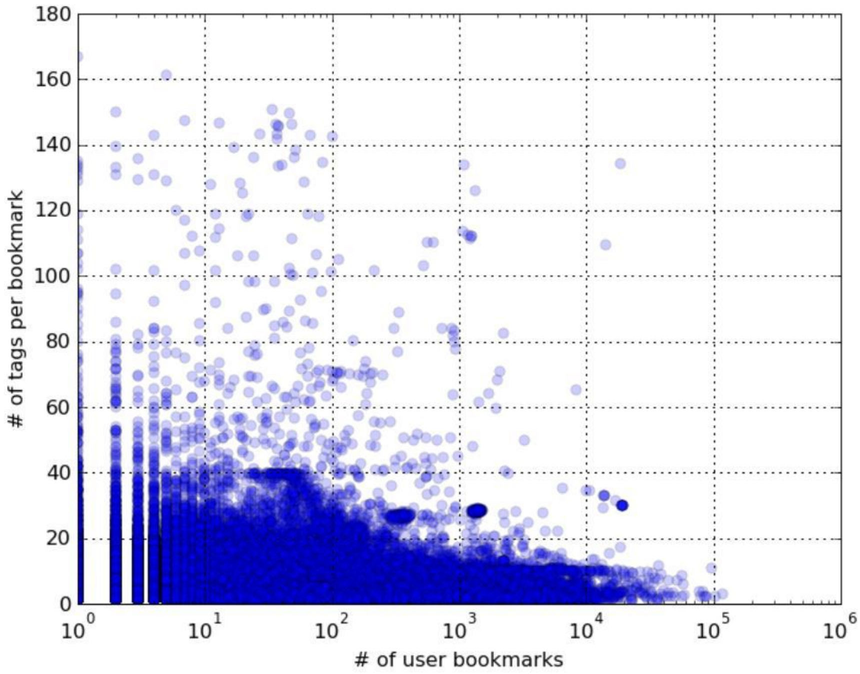


Figure 2.7: The number of user bookmarks compared to the average number of tags assigned by the user (*semi-log* scale). Some spammers try to increase the visibility of their content by assigning a very high number of tags.

Does order matter?

Research on collaborative tagging has so far only looked at the outcome of the tagging process, and less at the process itself. This comes to no surprise, since existing datasets, like $bibsonomy_{2009}$ or *citeulike*, do not provide any information about the actual position of a tag during the assignment process. Fortunately, tag positions are preserved within our Delicious dataset, and we find that the position of a tag correlates with its expressiveness. Figure 2.8 compares the entropy of tags by their position during the assignment process. The evaluation is based on the $delicious_{clean}$ dataset. Every point of the figure was calculated over one million consecutive bookmarking events, where every bookmark had to contain at least 3 tags in order to be considered in the evaluation.

Entropy is a measure for the uncertainty of a random variable and is

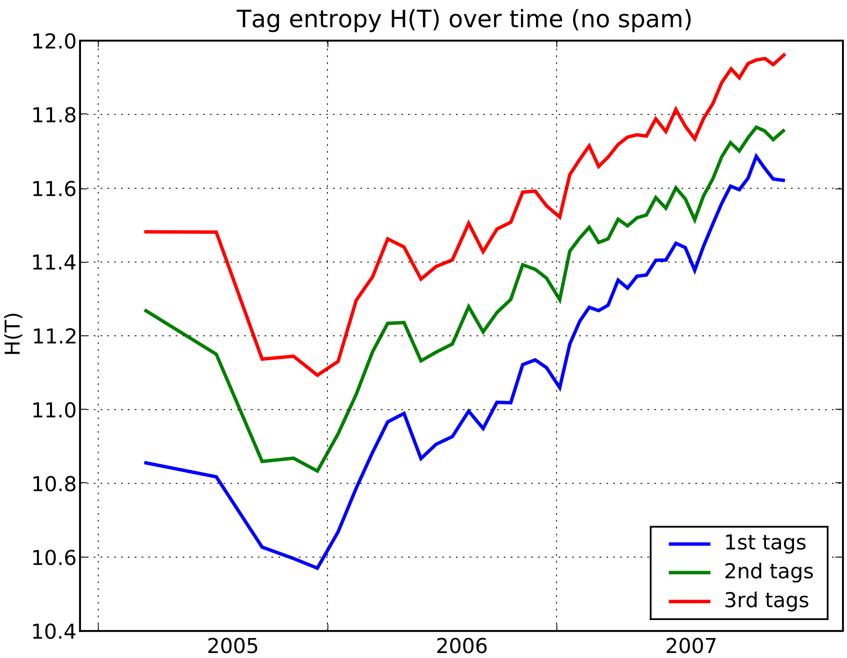


Figure 2.8: Tag entropy by tag position over time for $delicious_{clean}$. Each data point was calculated over one million bookmarks. Bookmarks with less than three tag assignments have been discarded to make results comparable. The plot shows a strong correlation between the position of a tag and the measured entropy. Users tend to assign common tags at the beginning of the labeling process and more specific tags later. This behavior remains surprisingly constant over all years.

defined as

$$H(T) = -\sum_{t \in T} p(t) \log_2 p(t). \qquad (2.1)$$

A low entropy implies that a tag distribution is dominated by very frequent tags, whereas more uniformly distributed tag distributions exhibit higher entropy. Figure 2.8 emphasizes the strong correlation between the position and the informativeness of a tag. It seems that users tend to assign common tags at the beginning of the labeling process and more specific tags later. This behavior remains surprisingly constant over all years.

We believe that there exist at least three potential explanations for this effect: (1) The tendency to label from broad to narrow could be a general behavioral pattern of humans that reappears in other domains. (2) The effect could also result from users' intention to categorize new content into a set of relatively constant categories. Adding frequent category tags in the beginning and content specific tags later would result in an increase in entropy as the one observed in Figure 2.8. (3) Finally, the observed relationship between tag position and entropy could be caused by the Delicious tag recommendation functionality. A tag recommender that recommends popular tags before less frequent ones combined with a tendency to select recommended tags in order of their presentation, e.g. from left to right as in Delicious, would produce curves similar to the ones of Figure 2.8.

The observation that the most descriptive tags do not necessarily appear first was previously reported by Liu et al. [LHY+09] for the Flickr service. The authors reason that the order of tags provides little additional information for tag-based retrieval solutions. Our results instead indicate that later tags are of higher expressiveness than their predecessors and thus of higher value for potential retrieval.

Do tag distributions reflect external events?

The distribution of tags within Delicious is not constant but changes over time and often significantly correlates with external events. Figure 2.9 presents the dynamics of 5 sample tags in 2007. As can be seen from the time series, the tagging trends reflect both the upcoming of new technologies, such as Google's Android which was announced in early November 2007 as well as periodic events, such as christmas. The delay between an external event and its echo on Delicious is generally marginal, which was also reported by [HKGM08]. Social bookmarking services can therefore provide a valuable and cheap resource for the detection, monitoring, and even prediction of real-world trends. We will present a method for tag-based trend detection in a folksonomy environment in Chapter 9.

Measuring the popularity of tags by the number of tag assignments will weight users by their activity. This will make such a measure vulnerable to spam. With the *diffusion-of-attention* concept presented in Chapter 4, we will introduce a more advanced measure of tag popularity that overcomes the problems of simple occurrence-based methods.

What structures emerge from collaborative tagging?

Thus far, our analysis focused on tagging patterns that emerge on a macroscopic scale accumulated over the entire community. However, the design of effective folksonomy recommenders requires a deep understanding about

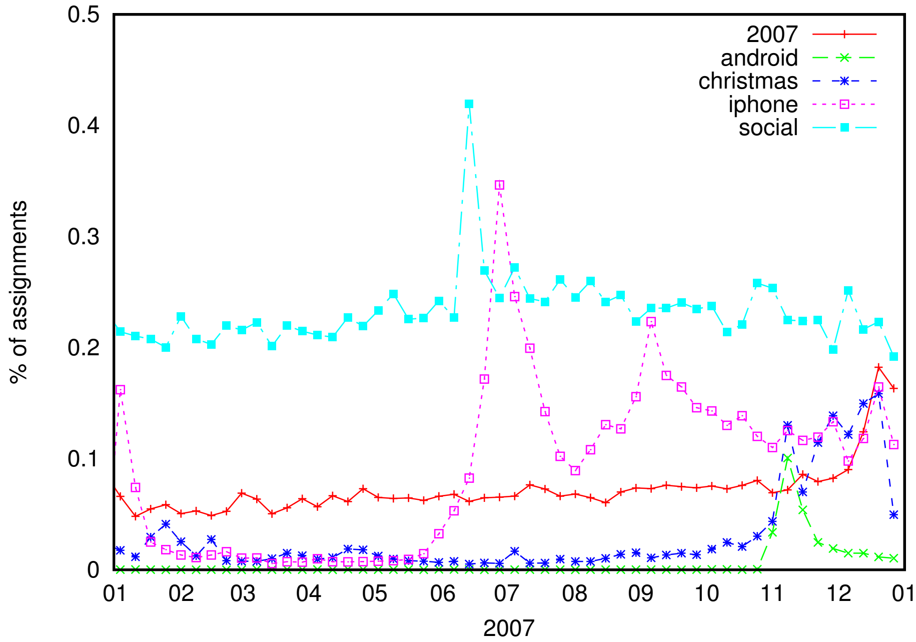
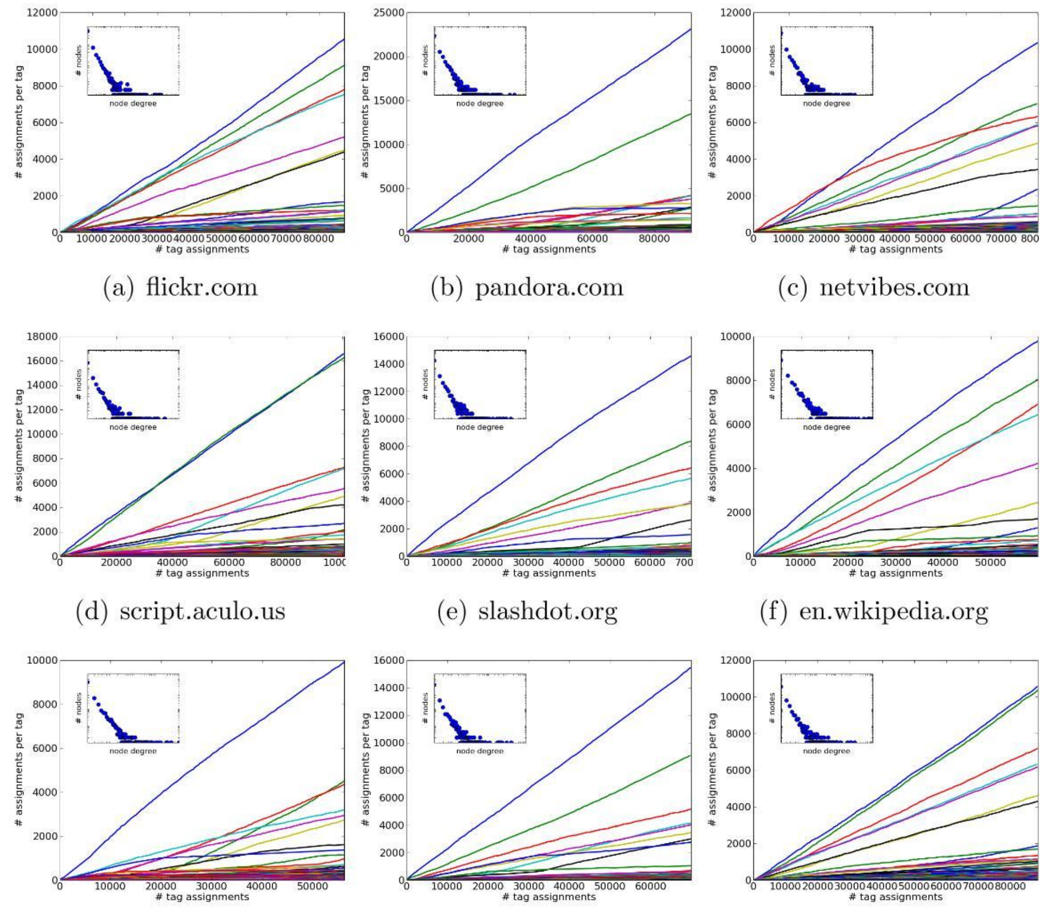


Figure 2.9: Popularity of 5 sample tags in 2007 as percentage of overall assignments. The distribution of tags within Delicious reacts to real-world events, including seasonal events such as "christmas" (see also Figure 4.3).

local patterns as well. We therefore continue our investigation by looking at the item vocabularies that emerge when many people freely label the same items.

Figure 2.10 illustrates how the tag distributions of the nine most popular URLs in the Delicious dataset develop over time. Each line corresponds to the frequency of one tag with respect to the overall number of tag assignments (TAS) to an item. We observe that the tag distributions of all URLs converge after already few bookmarking events, and do not change significantly thereafter. The collaborative labeling of content hence leads to the emergence of a characteristic, scale-free tag spectrum despite of user-level discrepancies. This stability implies that the folksonomy opinion about an item can be inferred, even if it has been bookmarked by only a subset of users. Tag-based item models will hence not require a constant retraining and remain valid over long periods of time.

The tag spectrum of all URLs is dominated by a minority of very frequent tags. In every distribution we find the most popular tag to be assigned in more than 15 percent of all cases, whereas the Top 5 tags appear in more



(a) flickr.com (b) pandora.com (c) netvibes.com

(d) script.aculo.us (e) slashdot.org (f) en.wikipedia.org

(g) youtube.com (h) last.fm (i) alvit.de/handbook

Figure 2.10: Tag distributions over time for the nine most popular URLs in the *delicious*$_{clean}$ corpus. Each line corresponds to the frequency of one tag w.r.t the overall number of tag assignments (TAS). The small plots show the corresponding node degree distributions on a *log-log-scale*, where the x-axis corresponds to the degree of a tag node, i.e. the number of times this tag was assigned, and the y-axis represents the number of tags of the same degree. Resources develop a characteristic and scale-free tag spectrum strongly dominated by very few tags. The stability of these spectra signifies a user agreement on the nature of an item, likely to be supported by the Delicious tag recommender. This convergence further underlines the potential of tags for content modeling. The frequency of popular tags exceeds even the one of an expected power-law behavior. (See Section 3.1 for details on node degree distributions and power-law characteristics.)

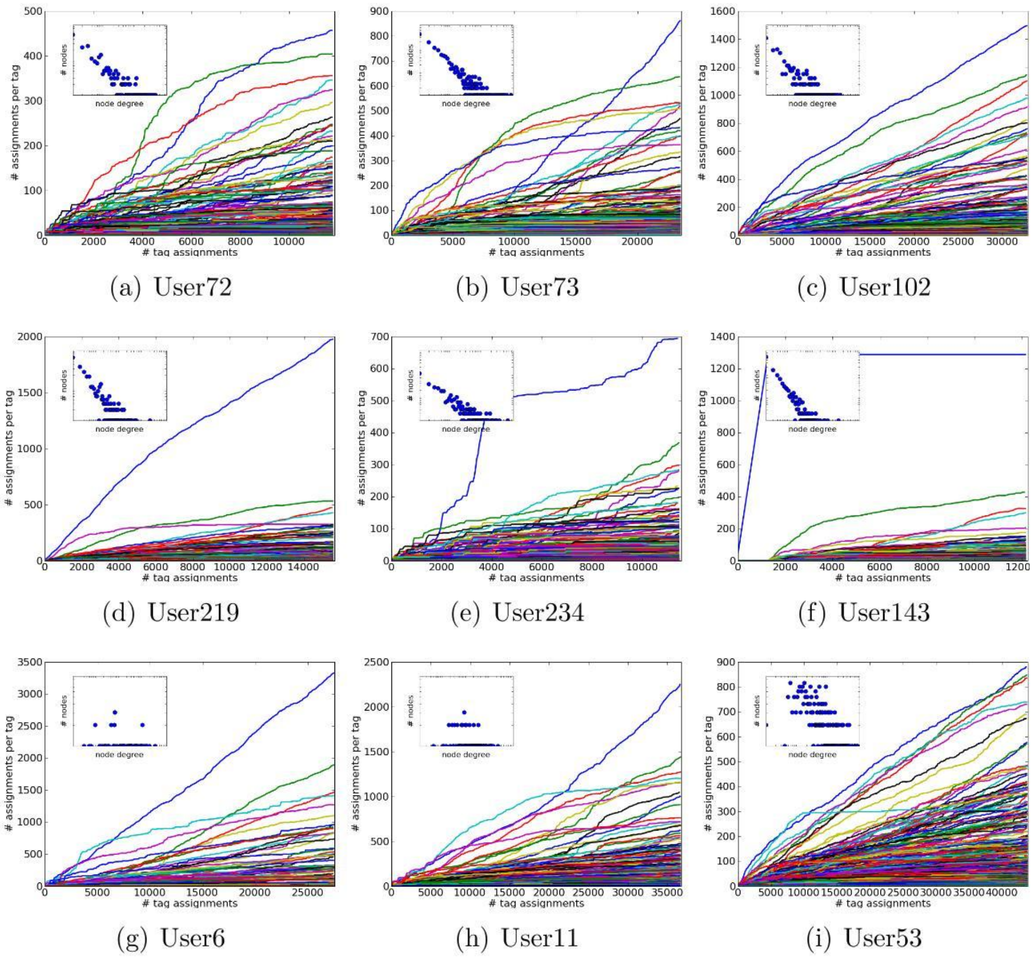
than 50 percent of all assignments. These high percentages imply that users generally share a common understanding about the nature of an item, an agreement likely to be supported by existing tag recommenders.

The small plots in Figure 2.10 display the corresponding node degree distributions of the final tag vocabulary on a *log-log* scale. Here, the degree of a tag denotes the number of times it was assigned to a URL. The y-axis of the plot represents the number of tags of the same degree. All plots highlight that, in contrast to the dominance of very frequent tags, the majority of tags appears only once. This inequality between dominant and infrequent tags seems even more pronounced than in power-law distributions, which have been comprehensively found in various types of networks including folksonomies [HJSS06a, LGZ08, OGC08]. We attribute this bias to the fact that, even though users agree on some characteristic resource labels, they additionally tend to assign their own sets of category tags. The diversity of these category tags results in the observed long-tail distributions of node degrees. These tags provide little insight from an information retrieval perspective. Instead, a minority of frequent tags may already be sufficient to comprehend the nature of an item. The previously mentioned criticism of tagging as too imprecise and dominated by personal tastes (see Section 1.1) therefore does not hold if many users tag the same resource. Moreover, the gradations in the frequencies of tags that occur in broad folksonomies provide a measure of relevance for each tag, with direct implications for information retrieval. The stability of item vocabularies is an important observation, since it can help in the design of more efficient and noise-reduced item models.

Even though we only present a sample of nine URLs in Figure 2.10, detailed manual analyses show that the stability of vocabularies is a general phenomenon for all items of a certain popularity. This stability therefore seems to be a direct consequence of accumulative effects that eliminate user-level heterogeneities. We exploit this characteristic when using items as intermediates in order to translate from personomy tags to the community vocabulary in Chapter 8.

How stable are personomies?

It is also interesting to look at how the tagging behavior of users develops over time. We plot the temporal development of nine exemplary personomies in Figure 2.11. The figure, first of all, supports the intuition that tag distributions are much less static from a user perspective than they are for items. In fact, we find a variety of behavioral patterns. The users in the first row of Figure 2.11 represent the most common archetypes of users with relatively stable tagging behavior. However, this stability is not comparable to the one



(a) User72 (b) User73 (c) User102

(d) User219 (e) User234 (f) User143

(g) User6 (h) User11 (i) User53

Figure 2.11: Tag distributions over time for 9 exemplary users of the $delicious_{clean}$ corpus. Each line corresponds to the frequency of one tag w.r.t the overall number of tag assignments (TAS). The small plots show the corresponding node degree distributions on a *log-log-scale*, where the x-axis corresponds to the degree of a tag node, i.e. the number of times this tag was assigned, and the y-axis represents the number of tags of the same degree. The users in the **first row** display a rather stable tagging behavior, even though their tag distributions shift in the long-term. The corresponding node degree distributions resemble a power-law. Most users of the dataset fall into this first category. The users of the **second row** show a very similar behavior, though their personomy is dominated by a single tag, which may indicate a dominant tagging strategy or impure data. The users of the **third row** show a very different tagging behavior, using only few tags.

of item vocabularies. Instead, we find personomies to react to shifts in a user's interests or in tagging strategies. However, at least for short periods of time these dynamics may be disregarded, such that tag-based user modeling remains generally possible. Unlike items, these user models will require constant updates in order to remain meaningful.

Looking at the node degree distributions for the first three users of Figure 2.11, we observe that the tags within their personomies exhibit power-law characteristics, such that the corresponding distributions appear as lines in the node degree subplots. These users reuse a small set of tags very often, whereas many tags are assigned only once. This reuse is likely not only due to recurring topics within a user's collection of bookmarks. Instead, it will be supported by a user's interest in a converging set of category tags for more efficient later retrieval. Users will avoid choosing synonyms, singular and plural forms, or tags from different languages for their category set. Tagging is thus not really "free" from the perspective of a user who is interested in organizing her bookmarks, since high randomness impedes efficient retrieval. On the other hand, we find the vast majority of tags within personomies to occur only a few times. These tags can reflect very rare interests of a user, but may also relate to item specifics, or simply represent misspellings or unintentional deviations from the standard set of category tags. We will later show that a user's interest in personal retrieval directly causes the emergence of individual tagging strategies in Chapter 8. A consequence of this heterogeneity is that even in cases where users obviously share the same interests by tagging the same items, this similarity is not necessarily reflected in their personomies.

Personomies also tend to be less biased towards very frequent tags when compared to item vocabularies. However, as illustrated by the examples (d)-(f) of Figure 2.11, this is not true for all users, and some personomies are dominated by a single tagging strategy. A detailed analysis of these users unveils that this effect may be due to the dominance of a concept, e.g. "gaming" (d), or a task, e.g. "dedicate" (e). However, some of these patterns also seem to be created artificially by existing bookmarking tools that always add certain tags, e.g. "imported" (f).

A very extreme form of controlled tagging is practiced by the users in Figure 2.11 (g) and (h). Both users have a very small vocabulary of about 100 tags that they constantly assign to tens of thousands of bookmarks. This results in very particular node degree distributions. Such rigorous tagging strategies may suit professional users who use Delicious to categorize web content into a predefined set of categories. However, these patterns could also indicate undetected spam users. Spam also explains the abnormal node degree distribution of the last user.

Fortunately, we find that the majority of personomies develops as illustrated by the first row of examples in Figure 2.11. This distributional conformity would generally encourage approaches that derive user similarities by calculating the distance between personomies [ZC08]. However, these methods will fail in cases where users develop individual tagging strategies. We will discuss the problem of inter-user heterogeneities in tagging and present a more sophisticated tag-based user model in Chapter 8.

How well do tag vocabularies cover future tags?

From a recommendation perspective, it is important to understand how well the tag vocabularies of items and users cover future tag assignments, and how this coverage correlates to the size of a vocabulary. This coverage has direct implications on scenarios, such as tag recommendation, where tags are generally predicted from past tag assignments by a user or to an item. We therefore plot the probability that an assigned tag is already part of an item's vocabulary or a personomy in Figure 2.12. We observe that the percentage of new tags covered by a user's personomy constantly increases as the personomy grows. The first 10 tags a user assigns reoccur in more than 50 percent of the following labels which is a further indicator of the stability of user interests in the short term. The coverage of the item vocabulary initially appears to be lower than the one of personomies. However, this is to be expected, since every user can assign more than one tag which reduces the chances of overlap during the initial phase. For both vocabulary types, we find that the first 30 tags will reappear in about 65 percent of the following labels, reaching a maximal coverage of more than 85 percent for large vocabularies.

The results of Figure 2.12 are significant for the tag recommendation scenario as they represent the maximum recall rates that purely item or user vocabulary-based recommenders can achieve given the tag assignments in the training data.

2.4 Chapter conclusions

This chapter provided an overview of the general motivations and emerging patterns in social bookmarking and tagging. Our analyses were based on a large-scale snapshot of the Delicious social bookmarking services. To no surprise, we found the interests of the Delicious community to be biased towards information technology-related content. This bias is reflected in the community's bookmarking and tagging behavior alike. Furthermore, our

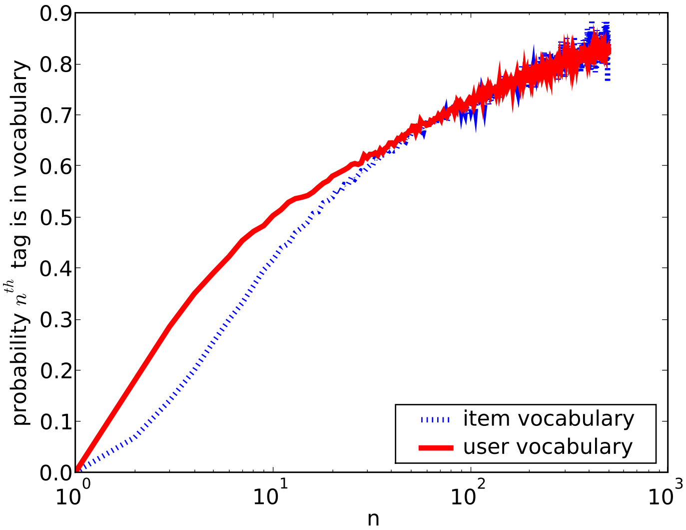


Figure 2.12: Probability that an assigned tag exists in a user's personomy or the item vocabulary as a function of the number of tag assignments (TAS) by each user and to each item (Delicious). The tag coverage constantly increases with growing vocabularies indicating converging tag distributions.

studies showed that users tend to return to previously bookmarked information sources. The number of bookmarks per domain thus does not necessarily reflect its popularity among the Delicious community itself. In Chapter 4, we will later use the diversity of a user's information sources for the design of a graph-based spam detector.

Looking at temporal bookmarking patterns, we observed that fame within Delicious is fugacious, and new URLs tend to be popular for a short period of time only. Recommender systems therefore should consider temporal aspects when determining the relevance of items. Furthermore, we identified different usage functions of tags. These functions directly affect the utility of a tag in a recommendation scenario. For instance, self-referential tags or individual category tags will be of low relevance in general search scenarios and for trend detection, but are of crucial importance in a tag recommendation setting.

One striking observation of our studies is the emergence of stable tag vocabularies for items which implies a community agreement on the most apparent characteristics of resources. This stability emphasizes the potential of tags for high quality and robust item models. On the other hand, personomies tend to be more dynamic, but still appear to be relatively stable in the short-term. User models that rely on personomies will thus be representative for future predictions, but are likely to require constant updates. The observed stability of item and user vocabularies is one of the motivations for the translational tag model presented in Chapter 8.

The next chapter will extend our analyses by investigating the structural properties of folksonomies from a network perspective. Many of the identified network properties will be of direct relevance for the later design of effective recommendation models.

Chapter 3

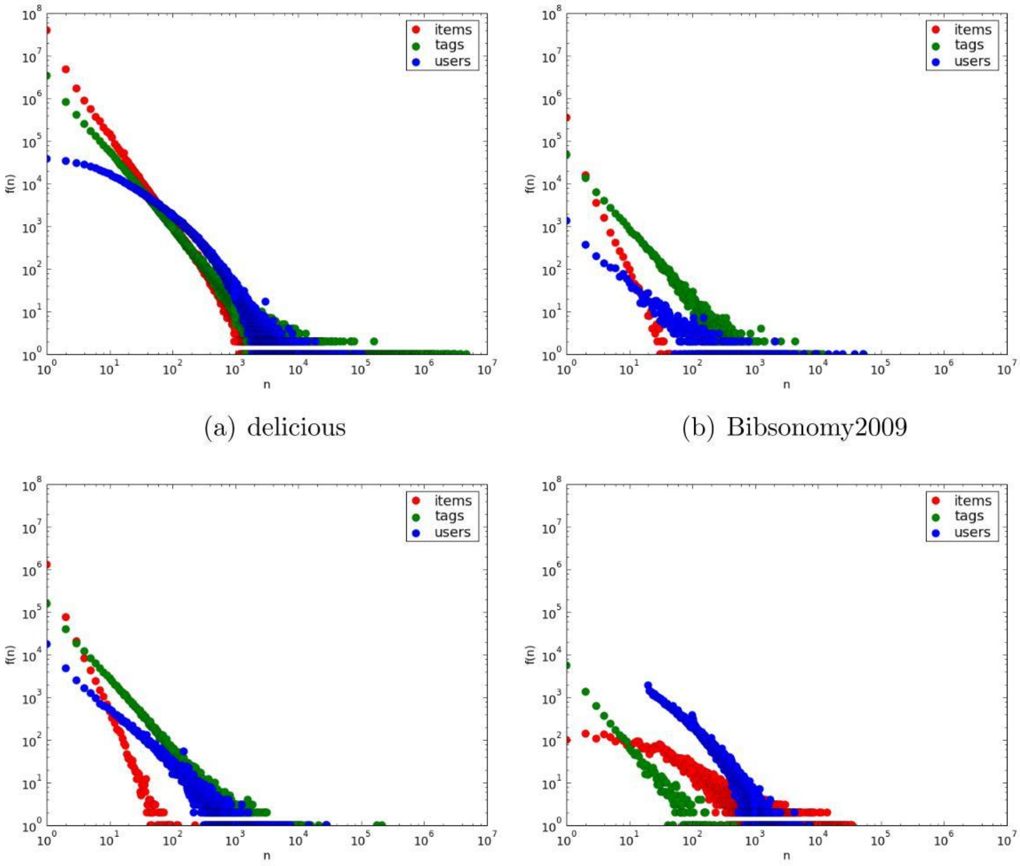
Network properties

This chapter discusses folksonomies from a network perspective. The discussion will focus on different structural and temporal network properties, such as node degree distributions, network growth, or divergence patterns. Previous literature on network analysis primarily focused on unipartite graphs, such as social networks, the World Wide Web, or co-citation networks, where only one type of node is present [New03, CF06]. On the other hand, folksonomies emerge from the interaction of three different types of nodes - items, tags, and users - and are therefore most naturally modeled as tripartite networks. In this chapter, we will study the network properties that characterize folksonomies from a graph perspective. The peculiarity of the identified properties will be of decisive importance for the design and the quality of folksonomy recommender systems.

3.1 Consistency

We start our analyses by investigating the node degree distributions of folksonomies. The degree of a node denotes the number of edges (TAS) it is part of. For folksonomies, this is equal to the number of bookmarks a node exists in, since every node can occur at most once per bookmark. The node degree of an item node hence denotes the number of times this item was bookmarked. Accordingly, the node degree for user nodes is equal to the size of their bookmark collection, and for a tag node it represents the number of times this tag has been assigned.

Our investigations confirm the findings of other folksonomy studies [Hey06, HJSS06a, OGC08, KJHS08, LGZ08], namely that item and tag node degree distributions in folksonomies follow a power-law, where very few highly connected nodes coexist with many sparsely connected nodes. According to



(a) delicious

(b) Bibsonomy2009

(c) CiteULike

(d) Movielens10M

Figure 3.1: Node degree distributions of items, tags, and users for different folksonomy datasets (*log-log* scale). The number of times an object occurs in a bookmark is plotted on the x-axis, whereas the y-axis represents the number of objects that occur x times. Item and tag occurrences in folksonomies generally follow a power-law the form $y(x) = Ax^{-\gamma}$, where few entities contribute much to the overall number of observations and many entities remain rare. Power-law distributions appear linear when plotted on a *log-log*-scale. User distributions in large-scale folksonomies often deviate from power-laws and follow a log-normal distribution. The differences in the *Movielens10M* dataset originate from a rather fixed set of rateable movies, the exclusion of users with less than 20 ratings, and a very low ratio of tagged movies of less than 0.25 percent.

[CF06] two variables are related by a power-law if the following equation holds:

$$y(x) = Ax^{-\gamma}. \tag{3.1}$$

Power-law distributions are a common phenomenon in natural and man-made environments and have been reported, for instance, for word occurrences in natural languages [MS99] or the degree of in- and out-links for pages on the World Wide Web [Kle99]. Networks that exhibit power-law conformity are also referred to as *scale-free* [BB03]. Plotted on a log-log-scale, a power-law distribution roughly resembles a line.

Figure 3.1 displays the node degree distributions of items, tags, and users for all evaluated datasets. The almost linear course of the degree distributions of item and tag nodes emphasizes power-law conformity. We observe that all datasets are dominated by a small ratio of very popular items and tags, whereas the majority of objects appears in the *long-tail*. This is emphasized when looking at the occurrence frequencies of URLs in the *delicious* dataset, as shown in Figure 3.1(a). Here, 39 percent of all bookmarks refer to the Top 1 percent of URLs, whereas 80 percent of all URLs appear only once in the corpus. Many of these long-tail URLs are related to spam, as we will discuss in Chapter 4. However, a further examination shows that the power-law compliance of items emerges independently of spam activity. In fact, we observe the same power-law patterns in the spam-filtered $bibsonomy_{2009}$ (Figure 3.1(b)) and $delicious_{clean}$ (not shown) datasets. Only the item distribution of the *movielens10M* dataset differs from a power-law. Here, the amount of movies with very few ratings appears underrepresented compared to popular ones. This can be explained by the rather static item set in this corpus, since movies are only added by the service provider.

The tag vocabulary of folksonomy users is highly standardized. Although there exist around six million tags in our Delicious corpus, we find that less than 700 tags, or 0.01 percent, account for 50 percent of all assignments, and the most frequently assigned tag ("design") occurs more than 4.9 million times. This convergence is likely aided by tag recommendation functionality which will tend to recommend popular tags. Additionally, other authors argue that users mimic the tagging behavior of other users, leading to converging tag vocabularies [Mat04]. At the end of the distribution, we notice that 55 percent of all tags in Delicious appear only once. The power-law conformity observed for tags corresponds to the frequency distributions of words in natural languages, where this characteristic is traditionally known as Zipf's law [Zip49]. However, as noted by [ALMP08], there exists no equivalent to stop words in social tagging.

Whereas a power-law distribution seems a feasible approximation for tag and item distributions, it does not fit users. Instead, as illustrated by Figures 3.1(a), (c), and (d), we find that folksonomies posses a higher number of *middle-class* users than we would expect in a power-law setting. For our Delicious corpus, parts of this deviating behavior are caused by the shortcomings of our crawling method that favored frequent users. However, similar patterns have been found by other authors for the distribution of user queries in search engines [KHS08a], citation networks [LAH07], click-streams [CF06], or words in the bible [BFK01]. These distributions can be described by a discrete truncated log-normal distribution also called the Discrete Gaussian Exponential or DGX [BFK01]. This distribution is of the form

$$P(x = k) = \frac{A(\mu, \sigma)}{k} \exp\left[-\frac{(\ln k - \mu)^2}{2\sigma^2}\right] \quad k = 1, 2, \ldots. \tag{3.2}$$

A is a normalization constant and μ and σ are parameters, such that the logarithm of a log-normal distribution is a Gaussian of type $N(\mu, \sigma^2)$. The DGX distribution extends the log-normal to discrete distributions [CF06], and has the power-law distribution as special case [BFK01].

Despite its non-conformity with a power-law, the user activity in folksonomies is far from being balanced. Instead, a few users generate the majority of posts. 22 percent of all bookmarks within Delicious originate from the Top 1 percent of users, the Top 10 percent contribute 62 percent of posts. These values exceed the ones reported by [HKGM08]. We assume that this difference is due to an increase in spam posts by time, as described in Chapter 4. Another portion of the increase likely stems from the bias of our crawling method.

One common consequence of power-law like node degree distributions is the *small-world* phenomenon [LH08]. Small-world networks are graphs where each node can be reached from another node traversing only few edges. These networks generally comprise a minority of highly connected nodes, so called *hubs*, which act as connectors for other nodes. This type of graph has been reported for social networks, the Internet, or protein networks [WS98]. The small-world character of folksonomies was first reported in [CSB+07]. The authors investigate the number of minimal hops that is required to get from one randomly chosen node of the folksonomy to another (*characteristic path length*). They report a characteristic path length of about 3.5 hops for Delicious. This path length remains stable during the growth of the folksonomy. As a result, an omniscient user who browses the folksonomy graph along its edges will only need to follow 3.5 links on average before she finds an object. This effort remains constant even when the folksonomy grows to millions of nodes. This observation emphasizes the small-world character of folksonomies. By definition, the characteristic path length in small-world networks has to grow less than logarithmically with the number of nodes in the network [New03]. Other authors report that the expected path length in real-world networks grows often much less than this value [BR04]. The reported constant path length for Delicious could be a consequence of the tripartite nature of the folksonomy and the high connectivity of many tag nodes.

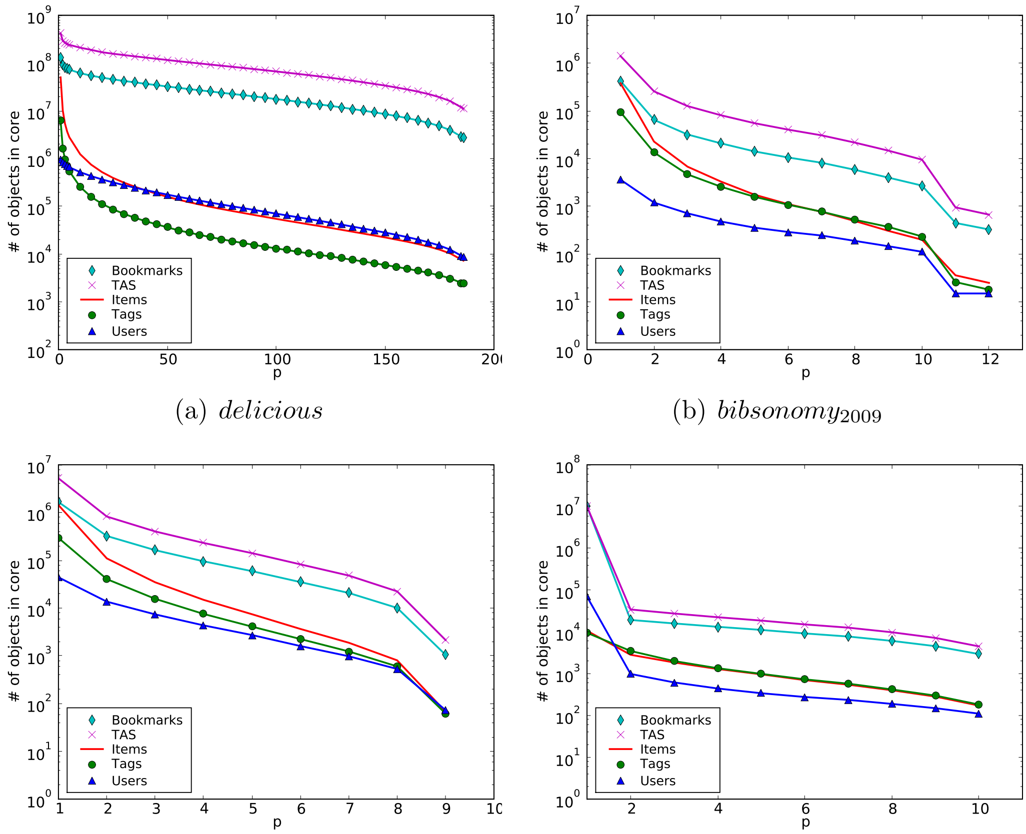
From a recommendation perspective, the small-word phenomenon implies that objects, such as a user or an item, will be connected within the folksonomy graph by only few hops. Their relation can thus be estimated without many intermediate steps. This high connectivity is mainly due to popular tags which emphasizes the potential of tags as connectors between users and items. This section analyzed the consistency of all node distributions independently. In the next section, we will instead explore how the different node types of folksonomies are connected to each other.

3.2 Connectivity

To investigate the connectivity among the nodes of folksonomies, we look at how quickly a folksonomy shrinks towards its core. The p-core of a folksonomy is the subgraph where each item, tag, and user node appears in at least p bookmarks [JMH+07]. This core can be determined by iteratively removing nodes that occur less than p times in the graph until no further node can be removed. Every folksonomy graph has exactly one subgraph that denotes its core at level p.

Figure 3.2 displays the effect of graph shrinking for all folksonomy datasets evaluated in this dissertation. All curves show a very similar three-phase behavior. Phase I sees a strong node decrease for low values of p. This is followed by a stable phase of logarithmic shrinking in Phase II, and completed by the collapse of the core (Phase III). The decay in Phase I for all object types follows a power-law with growing p, where a small increase of p removes the majority of nodes. One explanation for this behavior is the existence of small clusters within the periphery of the folksonomy graph that are not or only sparsely connected to the rest of the network. These clusters could represent small groups of users with deviating tagging or bookmarking behavior and will likely contain many spam users. This would correspond to the findings of Neubauer et al. [NWO09], who report a relatively low connectivity of spam users to the giant component[1] of the folksonomy network. However, we also experience power-law decays within

[1]The giant component of a graph is a connected subgraph that contains the majority of nodes [NWO09].



(a) *delicious* (b) *bibsonomy*$_{2009}$

(c) *citeulike* (d) *movielens10M*

Figure 3.2: Number of nodes, bookmarks, and TAS at different core levels (*semi-log* scale). The plot demonstrates that items and tags initially decay as a power-law with p, probably due to the removal of peripheral nodes of low or no connectivity to the main core. A similar behavior has been previously reported for other networks, such as the Internet [AHDBV05]. However, the decay slows down for increasing p values until the core decays logarithmically, forming lines on a semi-log scale. This behavior indicates a relative stability of the core during this phase, followed by the final collapse of the graph during the last p-core levels. The other datasets show similar behavior but at much smaller scale. For *movielens10M*, the limited number of tagged movies leads to a very strong graph reduction when going to $p = 2$. The network then behaves as the other datasets.

the spam-free *bibsonomy*$_{2009}$ dataset. This shrinking behavior is not unique to folksonomies, and the same rapid decay was reported, for instance, for the Internet graph [AHDBV05]. For the *delicious* dataset we observe that the initial pruning mainly removes very infrequent items and tags, such that the overall number of bookmarks and TAS remains rather unaffected. The differing user curve for Delicious probably results from the shortcomings of our crawling method, since other datasets seem unaffected.

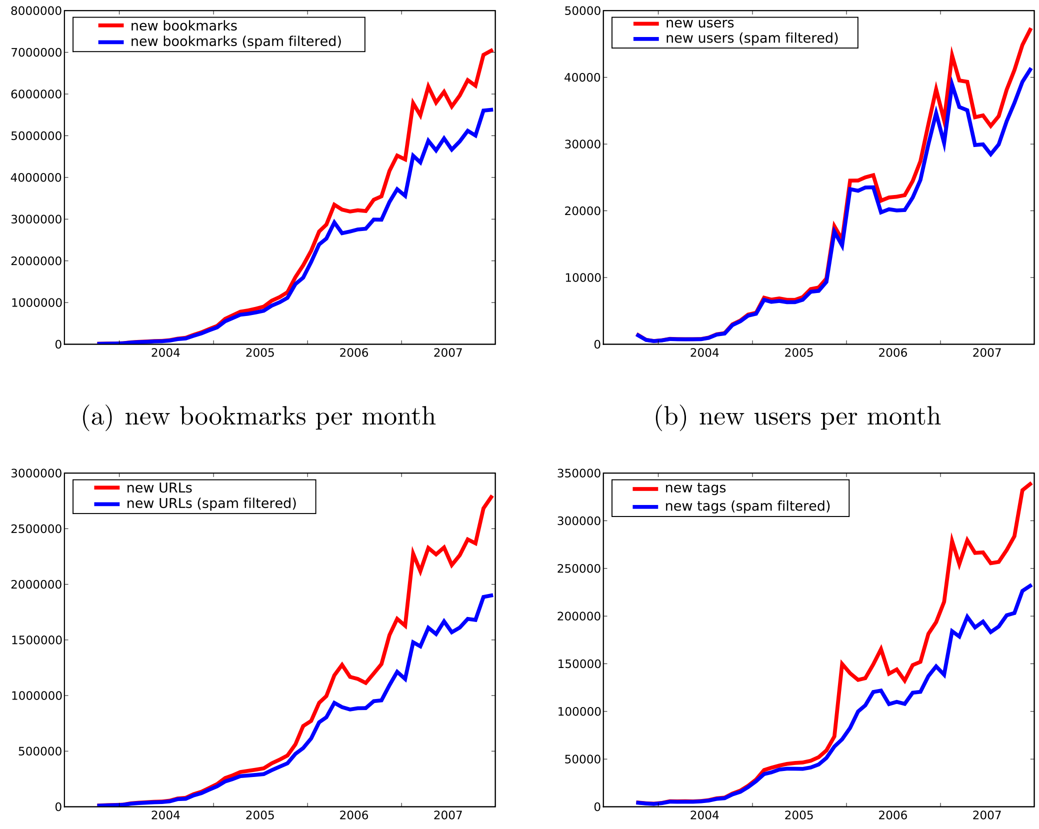
The initial power-law decay is followed by a stable and less drastic shrinking period. During this second phase, we find the core to shrink only logarithmically with growing values of p. The shrinking process appears remarkably stable forming lines on a semi-log scale. This stability probably means that the remaining nodes exhibit a high inter-connectivity after less connected peripheral nodes have fallen victim to the initial reduction steps. This second phase of stability ends with the collapse of the graph in Phase III. The main core of the *delicious* graph consists of $8,766$ users, $7,308$ items, and $2,406$ tags concentrated in $2,720,926$ bookmarks. The most popular tag in this core is "css", which again highlights the focus of the Delicious community on topics related to web development. The entire *delicious* graph collapses for $p > 186$. All other folksonomy graphs collapse much earlier.

A surprising result for *delicious* is the identical shrinking behavior in the number of item, tags, users, bookmarks, and TAS during the second phase of relative stability. This signifies a stable ratio of the sizes of all sets of objects, TAS, and bookmarks, which in turn implies that all sets decay at the same rate. Accordingly, we find the average number of tags per bookmark to remain surprisingly stable. This means that the number of tags per bookmark does not affect the probability that a tag will be pruned.

Yet another remarkable observation is the universality of results over all folksonomy datasets. Even though the plots of the smaller datasets display higher variability, we find that all datasets display the same three phases of drastic shrinking, steady decay, and collapse. These phases therefore seem to be characteristic for folksonomy networks.

3.3 Growth

Delicious is a fast growing on-line community, as shown in Figure 3.3. This growth process finds its expression not only in a continuously increasing number of users and postings but also in constantly growing URL and tag sets. According to our dataset, there were at least $47,429$ newly appearing Delicious users in December 2007, more than in any previous month. This exponential expansion is accompanied by a constantly growing bookmarking



(a) new bookmarks per month

(b) new users per month

(c) new URLs per month

(d) new tags per month

Figure 3.3: The monthly growth of Delicious by posted bookmarks, new users, new URLs, and new tags. The blue (lower) line shows the growth patterns for the $delicious_{clean}$ dataset when spam is filtered as described in Section 4.3.

rate, culminating in about 7.5 million new bookmarks per month for the last months of our dataset. We also observe exponential growth patterns in the numbers of known URLs and tags. Similar growth patterns, though at lower scale, can be found for the $bibsonomy_{2009}$ and *citeulike* datasets. The expansion of the MovieLens community differs, since new movies are added by the provider, resulting in a rather static set of movies.

Figure 3.3 further emphasizes the contribution of spam to the expansion of Delicious. Applying the spam filtering method presented in 4.3, we observe a high ratio of previously unknown URLs being posted by assumed spam users. This comes to no surprise, since spammers tend to promote sites that "real" users would not bookmark. Furthermore, we find spam to generate many new tags. This may be due to spammers that create tags randomly or assign keywords from less present languages.

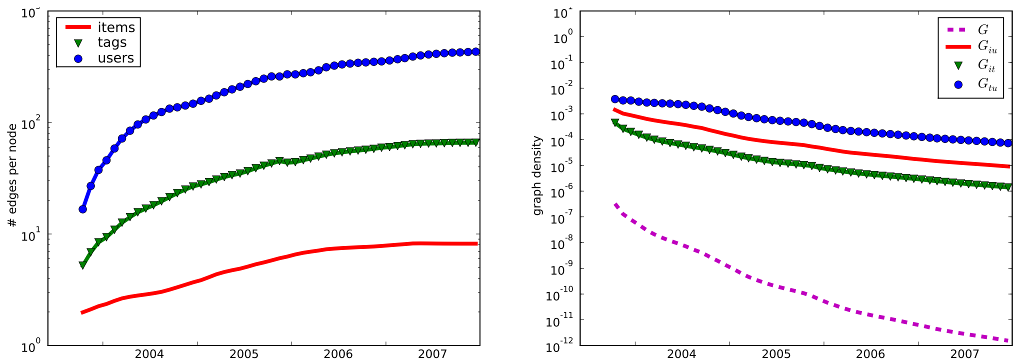
3.4 Density

A consequence of the expansion of folksonomies is a steady increase in the average number of edges per node. This behavior implies that the number of tag assignments (new edges) grows faster than the number of nodes, and can be observed independently of the node type. The average knowledge about a node thus increases over time, as visualized by Figure 3.4(a). Despite this observation, we find the overall graph density of folksonomies to drastically decrease during this expansion process, as highlighted by Figure 3.4(b). The density of a graph is given by the ratio of existing edges to all possible edges. The density of the tripartite undirected folksonomy graph G can thus be calculated as:

$$D(G) = \frac{|E|}{|I| * |T| * |U|}. \tag{3.3}$$

The observed decrease in graph density over time implies that the combined growth of all sets of nodes exceeds the increase in the number of tag assignments, i.e. edges. Collapsing G to the bipartite subgraphs G_{IT}, G_{IU} and G_{TU} reduces this density loss by some orders of magnitude. However, the overall growth of the community also increases the sparsity in these graphs. This tendency seems to be a common phenomenon for folksonomies and is also observed for the two other communities with open item sets, Bibsonomy and CiteULike. For MovieLens, one of the best researched recommendation datasets, we find a steady densification of the user item graph G_{IU} over time instead. We attribute this effect to the rather static set of movies within *movielens10M* and the artificial pruning of infrequent users.

Sparseness will generally hinder recommendation, since it represents the relation of known to unknown. For example, an increase of sparsity in the graph G_{IU} means that an item recommender needs to estimate more user preferences from a lower ratio of trained samples. Whereas sparsity decreases over time in classical recommendation settings, such as movie recommendation, we find that the relative knowledge about the interaction between users and items decreases continuously in those folksonomies that let users freely add items. Tags can help to reduce these sparsity problems, since they act



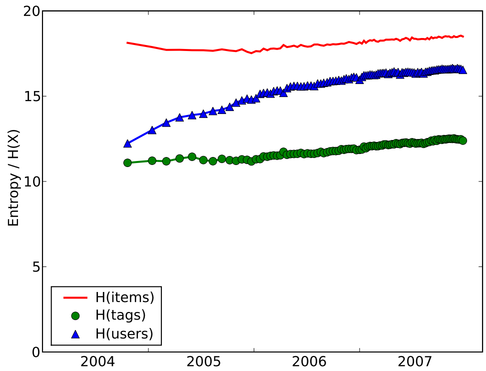
(a) edges per node (b) graph density

Figure 3.4: Average node degree (a) and graph density (b) for the *delicious* dataset plotted over time for (log scale). The number of edges per node (node degree) grows faster than the number of nodes such that the average node degree increases by time (a). However, the graph density - the number of existing edges compared to the number of possible edges - constantly decreases for the tripartite folksonomy graph G and even for the bipartite subgraphs G_{IT}, G_{IU}, and G_{TU} (b). This is equivalent to an increase in data sparsity over time, making recommendation more difficult. The user related curves plotted here are of low validity for the entire Delicious network, since our crawling method favored frequent users. The average node degree and density for these curves will be lower than presented. The impact of spam on these trends is negligible.

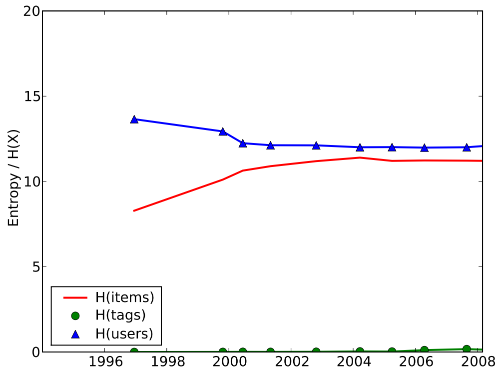
as connectors between users and items. As a result, we observe that recommendation models that integrate tags, such as the hybrid model introduced in Chapter 7, achieve higher accuracy.

3.5 Entropy

We previously observed that the node degree distributions in folksonomies tend to form power-laws with a small ratio of highly connected nodes. However, we did not investigate how this node concentration changes over time. As described in the previous chapter, one common way to measure the concentration of a distribution is to look at its entropy. The entropy of a distribution describes the average number of bits needed to encode an object.



(a) Delicious



(b) Movielens10M

Figure 3.5: Entropy of node distributions for the $delicious_{clean}$ and *movielens10M* datasets over time. Each point corresponds to the entropy of one million observations. To make results comparable, only the first tag assignment of each post was considered. Posts without tag assignment have been discarded. Whereas the entropy of items and tags within Delicious remains relatively stable over time, we find the user activity to become more uniform. This is likely caused by the constant arrival of new users that is not (fully) compensated by an increased activity of old ones. The *movielens10M* dataset shows a different behavior. User and item entropies converge. Items are much less random than in Delicious, simplifying the recommendation task.

This value will be large for uniform distributions, i.e. distributions of high *randomness*. Strongly inhomogeneous power-law distributions should require less encoding efforts instead. From a recommendation perspective, objects from distributions with high randomness (entropy) are harder to predict than objects from a less random distribution.

Figure 3.5 displays the entropy of all node degree distributions over time for the $delicious_{clean}$ and the *movielens10M* datasets. Each point corresponds to the entropy calculated over one million consecutive events. In order to make the plots comparable, we only consider the first tags assigned to bookmarks and disregard bookmarks without a tag. The $delicious_{clean}$ plot again emphasizes that tags are of higher density than URLs. The entropy for both types of nodes constantly increases over time, though only marginally. The general mixture of frequent and infrequent nodes within these distributions thus remains relatively constant. On the other hand, the user distribution becomes more uniform with the expansion of Delicious. The constant arrival of new users is hence not compensated by an increased activity of existing ones. This tendency is most likely related to the previously observed non-power-law conformity of the user node degree distribution (see Figure 3.1).

Contrary to the results from Delicious, we find item and user entropies in MovieLens to converge a period of a few years. This convergence takes place at a much lower level than for the $delicious_{clean}$ dataset. The URL recommendation task is hence much more challenging than movie recommendation.

In this section we calculated the entropy of node distributions for different periods in time. However, we did not yet explore to what degree distributions actually change. We will therefore evaluate the divergence of node distributions in the next section.

3.6 Divergence

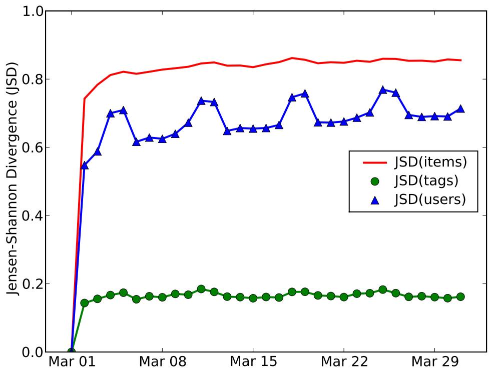
Recommender systems implicitly assume that future recommendations can be inferred from historical data. Items are recommended because users liked them in the past or because they are similar to past preferences. But what should be recommended if future preferences within a community vary drastically from past ones?

To measure distributional changes between consecutive points in time, we introduce the Jensen-Shannon divergence [MS99]. The Jensen-Shannon divergence is a symmetrized and smoothed version of the Kullback-Leibler divergence [MRS09] and thus closely related to the previously discussed entropy measure. The Kullback-Leibler divergence denotes the dissimilarity between two distributions P and Q. The divergence of P with respect to Q is calculated as the difference between the cross entropy of both distributions and the entropy P:

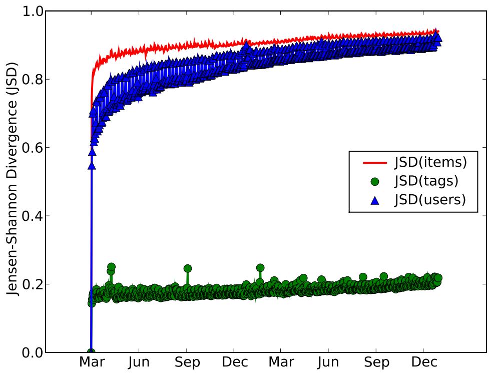
$$\begin{aligned} D_{KL}(P||Q) &= H(P,Q) - H(P) && (3.4)\\ &= \sum_i P(i) \log \frac{P(i)}{Q(i)}. && (3.5) \end{aligned}$$

The Kullback-Leibler divergence is neither symmetric ($D_{KL}(P||Q) \neq D_{KL}(Q||P)$), nor normalized, which makes it inappropriate for growing distributions. The Jensen-Shannon divergence addresses these shortcomings. We calculate the Jensen-Shannon divergence, which is symmetric and normalized to $0 \leq D_{JS} \leq 1$, based on D_{KL} as:

$$D_{JS}(P||Q) = \frac{1}{2} D_{KL}(P||M) + \frac{1}{2} D_{KL}(Q||M). \qquad (3.6)$$



(a) one month



(b) 22 months

Figure 3.6: Jensen-Shannon divergence for daily item, tag, and user distributions in the $delicious_{clean}$ dataset w.r.t the respective distributions of March 1st, 2006. (a) shows the divergence for all days of March 2006, whereas (b) looks at long-term effects. Already after only one day we find a different mixture of items and users (a). This divergence constantly increases over time. On the other hand, the tag distribution remains relatively stable. However, zooming to larger intervals (b) shows a steady divergence also in the mixture of tags. The peaks of the user curve correspond to weekends when the user-base seems to shift. This shift is reflected in the respective item and tag distributions. The peaks of the tag curve in (b) are probably due to incomplete spam filtering.

Here, M is simply the mean of both distributions ($M = \frac{1}{2}(P+Q)$). $D_{JS} = 0$ implies that two distributions are equal, whereas $D_{JS} = 1$ stands for maximal dissimilarity, i.e. that the distributions do not share any events.

Figure 3.6 plots the Jensen-Shannon divergence for daily item, tag, and user distributions in the $delicious_{clean}$ corpus with respect to the respective distributions of March 1st, 2006. Plot 3.6(a) highlights the divergence during the first month, whereas Plot 3.6(b) illustrates long-term effects. The plots show that the mixture of bookmarked items in Delicious drastically changes within a single day, shifting even more during consecutive days. This supports our previous finding that attention within Delicious is short-lived. An item recommender in such an environment therefore has to be highly up-to-date or has to identify items of longer life.

The divergence of users is less drastic, though still high. From a recommendation perspective, this signifies that the users a recommendation model was optimized for are not necessarily the same users recommendations will be required for. Figure 3.6(a) further highlights that user participation differs between working days and weekends, when the overall bookmarking rate drops. This changing community structure, combined with an interest shift during weekends, also influences which items are bookmarked and which tags are assigned. However, tags and items shift much less on weekends than the user distribution. Nevertheless, incorporating such temporal patterns into recommendation models will improve the quality of suggestions, especially in cases where users exhibit a different bookmarking behavior during week days.

Of all node types, we find the tag distribution to vary the least with time. The mixture of interest fields among the Delicious users is thus rather stable, at least from a global perspective. However, zooming to longer intervals (Figure 3.6(b)) unveils interests shifts that are reflected by changing global tag vocabulary. The relative stability of tags once more emphasizes their potential for more robust recommendation models. Models derived from tags will be of longer relevance than purely usage-based, i.e. collaborative filtering models. Our findings also imply that is easier to predict future tags based on past knowledge than to predict users and items.

3.7 Chapter conclusions

In this chapter, we studied various network properties and their peculiarity in folksonomies. Our results confirmed previous findings by other authors [Hey06, KHS08b], who reported items and tags within folksonomies to follow power-laws, whereas user participation is best represented by a log-normal distribution. One of the most important observations of our studies is that tags tend to be highly concentrated. The most popular tags occur orders of magnitude more frequently than the most frequent items or users. These popular tags act as hubs between the other object types. As a result, we found the characteristic path length between two random nodes of the Delicious folksonomy to remain small even if the network grows to millions of nodes. The similarity between two objects, e.g. a user and an item, can hence be calculated along only few intermediate nodes.

Furthermore, our findings showed that the openness of folksonomies makes item recommendation in folksonomies a more demanding task than classical movie recommendation. While the information density in movie recommendation increases over time, we observed a contrary behavior for all bipartite folksonomy subgraphs as well as for the folksonomy tensor itself.

Moreover, node distributions in folksonomies exhibit a constantly growing randomness compared to stable entropy patterns in MovieLens. The recommendation task in folksonomies is therefore more challenging than for other well-known recommendation settings.

Another interesting result is the instability of item and user distributions in folksonomies over time. Successful item recommenders will have to incorporate temporal aspects and will require constant updates in order to reflect the latest community interests. On the other hand, we observed that tag distributions are much less dynamic, which further emphasizes the potential of tags for the design of more robust user and item models.

Throughout this chapter, we saw that spam has a considerable impact on some of the examined network properties. Accordingly, spam needs to be filtered before any sophisticated recommender systems can be designed. The next chapter therefore explores the characteristics of spam within folksonomies and describes potential countermeasures.

Chapter 4

The problem of spam

Similar to applications such as email, spam also severely impacts folksonomies. An initial analysis of the *delicious* corpus revealed a frequent occurrence of bookmarks that originated from spam and were posted by automated mechanisms. Among the Top 20 most active Delicious users we found 19 users of apparently non-human origins, posting hundreds of thousands of URLs that belong to very few domains. These 19 "users" alone account for $1,321,316$ bookmarks, approximately 1 percent of the entire corpus. Unfortunately, this result comes to no surprise, since Delicious offers an interface for remote postings, and URLs that appear on Delicious have the potential of reaching thousands of users. Additionally, spammers may hope to augment the search engine ranking of their sites if their URLs appear on as many pages as possible[1]. Delicious is not the only bookmarking service vulnerable to spam. The authors of [KHS08a] report that among the $20,000$ users of Bibsonomy they manually identified $18,500$ as spammers. These spammers had posted 90 percent of all bookmarks.

If not detected, spam activity strongly influences the effectiveness and credibility of recommender systems. Fortunately, we find the intentions and behavioral patterns of spam users to differ drastically from those of non-spam users, allowing the effective detection of spam at an early stage. Section 4.1 characterizes spam in folksonomies on the example of Delicious and explores the data patterns that emerge from spam activity. The next section then discusses general approaches to the problem of spam detection and also introduces the *diffusion-of-attention* concept for efficient spam suppression.

[1] Social bookmarking sites, such as Delicious or Bibsonomy, generally mark outgoing links with the "nofollow" attribute, such that Google and most other search engines will not consider these links in their rank calculations. However, users that browse the folksonomy graph to discover new content will likely encounter highly linked spam if no filtering is performed.

Finally, Section 4.2 describes the graph-based filtering rules that were applied to clean the *delicious* corpus and explores how the filtering affects the previously described data patterns.

4.1 Characteristics of folksonomy spam

The behavior and thus the potential impact of spam creators differs. One Delicious user labeled all her $7,780$ bookmarks with the exact same six tags. All posts referred to the same blog site. Another spammer added over $10,000$ posts, linking to only few web portals. All posts had been labeled with more than 100 tags, presumably in order to make the related sources more visible to Delicious users and search engines alike. Furthermore, we encounter bulk uploads, a phenomenon also reported for Flickr by [DKM+06], where users upload thousands of bookmarks within minutes and rarely actively contribute thereafter. We also observe that spammers try to increase their impact by using multiple user accounts. The most prolific Delicious spammer posted at least $538,045$ bookmarks to only 5 domains using more than 10 accounts.

We observe that spammers tend to exhibit one or more of the following characteristics:

Very high activity. Automated posting routines may reach much higher participation rates than human users.

Few domains. Spammers will try to promote their own domains. As a result, we find that many domains receive a very high number of postings coming from only a few users, as previously describe in Section 2.2.

High tagging rate. Some spam users tend to label their bookmarks with an exorbitant number of tags in order to increase their visibility. In consequence, we observe higher average tag rates for these users. See Section 2.3 for details.

Very low tagging rate. Other spam users do not tag at all, but constantly upload bookmarks without any tags. This explains the high decrease in the ratio of unlabeled bookmarks when spam is filtered from the *delicious* dataset, as reported in Section 2.3.

Bulk posts. Bulk uploads generally imply automated postings. However, bulk uploads are not always spam. Instead, automated postings may also originate from human users, e.g. if a user synchronizes her local bookmarks or bookmarks from different bookmarking services by using existing software tools.

No retrieval intention. Spammers are interested in the distribution of their content, but do not need to access it later. The missing retrieval intention is likely to result in anomalous tag distributions for spam users, such as a very low tag entropy for spammers that always assign the same tags or an above average entropy for spammers that assign tags randomly. Furthermore, our analysis showed that the majority of personomies will develop a power-law conform tag distribution, whereas this phenomenon is rather unlikely for spammers. Examples for normal and abnormal tagging patterns are given in Figure 4.2. See also the discussion of personomy development in Section 2.3.

No tag bundles. A consequence of the missing retrieval intention is that spammers have no motivation to structure their tags into groups, for instance by using the Delicious tag bundle functionality.

Frequent use of spam tags. Spammers tend to promote content of a certain type. This is reflected by the tags they choose as entry points to their sites. The frequent use of tags such as *porn*, *buy*, *adult* or *cheap* is generally a strong indicator of spam activity, and spam detectors trained on these spam tags have been shown to produce very good results in [GK08].

Low connectivity. Spammers generally try to promote resources nobody else would bookmark. This results in a low connectivity within the user-item graph G_{IU}. Figure 4.1 depicts the distribution of spam and non-spam users over different component types [NWO09]. Figure 4.1(a) indicates that spammers have a much lower likelihood to share items with the community (giant component). Instead, spam users cluster together to form so called next-largest components that are not connected to the giant component. Spam users within a next-largest component are likely to originate from the same source. However, there also exists a large group of isolated non-spam users. Users that fall into this group have generally posted very few bookmarks. On the other hand, the connectivity of the user-tag graph provides less distinctive information, as is shown in Figure 4.1(b).

Combinations of the above. In most cases, we find a combination of the above characteristics. For instance, as previously described in Section 2.3, we observed that many users with large collections of bookmarks are also very active taggers and can thus easily be identified as spam.

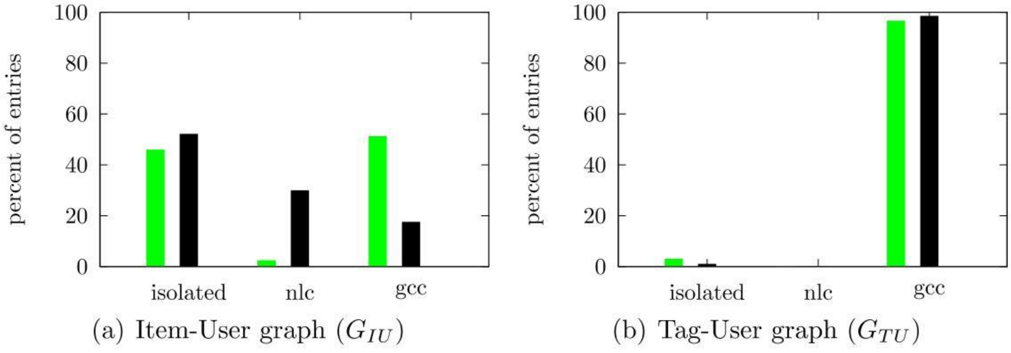


Figure 4.1: Distribution of spam (black) and non-spam (green) users over different component types for the *bibsonomy*$_{spam}$ dataset. Isolated components containing only a single user, next-largest components (nlc) containing more than one user, but not being the giant component, or the giant component (gcc). Spam users are less likely to be connected to the giant component within the item-user graph G_{IU}, but tend to form next-largest components. On the other hand, the connectivity patterns found within the tag-user graph G_{TU} provide less distinctive information [NWO09].

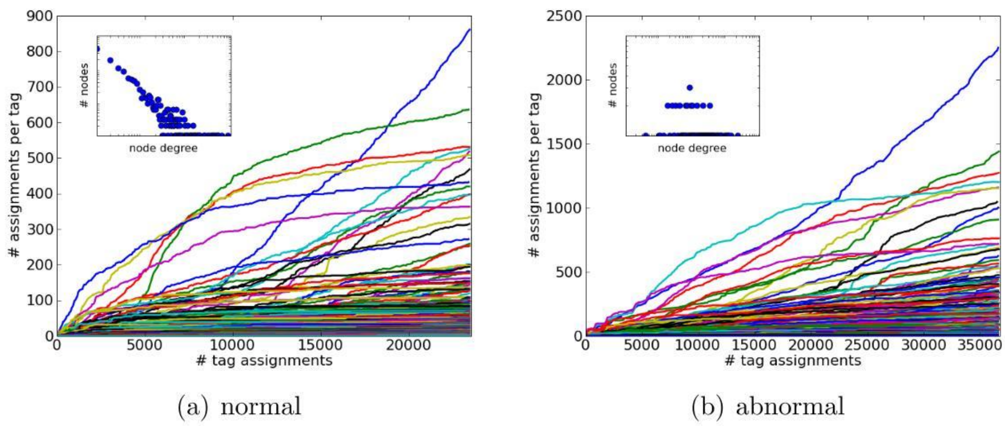


Figure 4.2: Spammers are not interested in a later retrieval of their bookmarks and therefore develop personomies with abnormal tag statistics, e.g. no power-law conformity. See Section 2.3 for details on the evolution of personomies.

The above characteristics make spam activity generally distinguishable from normal bookmarking behavior. We discuss possible spam detection methods in the following section.

4.2 Spam detection methods

The presence of spam may cause highly misleading results if folksonomies are exploited in scenarios, such as recommendation, trend detection, or tag-based ontology creation. In these cases, spam filtering should precede every model building process. This section describes the two most common principles to spam detection: text-based and graph-based spam detection. Furthermore, it introduces the *diffusion-of-attention* concept that drastically reduces the impact of spam.

Spam detection as text classification

Text-based spam detectors analyze textual features of a user's bookmarking activity to distinguish spam. These classifiers are thus similar to spam detectors known from the email and blog domain. In social bookmarking systems, the most prominent source for text-based classifiers are tags. The winning team of the ECML PKDD 2008 Spam Detection Challenge combined user tags with other textual attributes of posted Bibsonomy items, e.g. the URL or the title of a publication, in order to construct user centric feature vectors [GK08]. The authors then trained an extended version of a Rocchio text classifier [MRS09] to learn spam from non-spam users.

The authors of [KHS08a] use manually defined black lists of tags to determine the *spamminess* of a user. Further, they show that certain patterns in a user's profile data, e.g. the source of the email address, may already indicate spam. Moreover, they find a strong correlation between user names and email addresses that include numerical characters and spam. Based on these and additional features, the authors train a Support Vector Machine (SVM) [Bis07] to determine the boundary between spam and non-spam users.

Even though text-based spam classifiers were shown to perform well in many settings, they can be easily tricked, e.g. by avoiding typical spam tags. Those adapting spammers may still be discovered using graph-based detection methods.

Graph-based spam detection

Graph-based spam detection methods detect spam users or posts based on abnormal connectivity patterns within the folksonomy graph G. These abnormalities are likely to occur, since spam users generally post content that is of little interest to normal users. In [NWO09], we designed and evaluated spam detectors for the Bibsonomy bookmarking service which were based on structural features only. Our results state that many spam users can be detected because of their low connectivity with the giant component in G_{IU} (see also Figure 4.1). Moreover, our experiments show that the tag-centric connectivity patterns within G_{IT} provide less distinctive information. This means that, despite distributional differences, the personomies of spam and non-spam users tend to overlap. However, detection results could be improved still by combining usage and tagging patterns and looking at connectivity patterns in the hypergraph G.

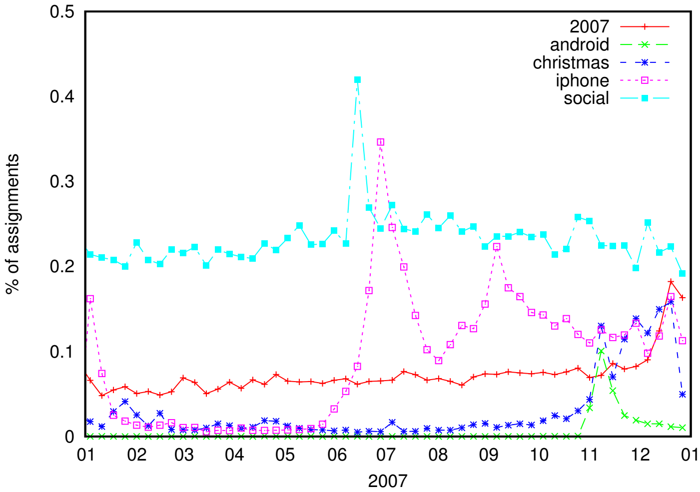
Similar to graph-based recommendation, graph-based spam detection does not depend on the underlying content type. This makes these methods especially attractive in scenarios where content features are hard to extract or analyze, e.g. for movies, music, or photos. Furthermore, tricking graph-based detectors will generally require higher effort from spammers, since their abnormal connectivity patterns directly result from their motivation to promote uncommon content.

Even though we find that our graph-based approach performs well for users with many bookmarks, it fails for new users, where little information yet exists. As was shown by [KHS08a] and [NO08], these *cold-start* problems can be reduced by combining text and graph-based features into a unified detector.

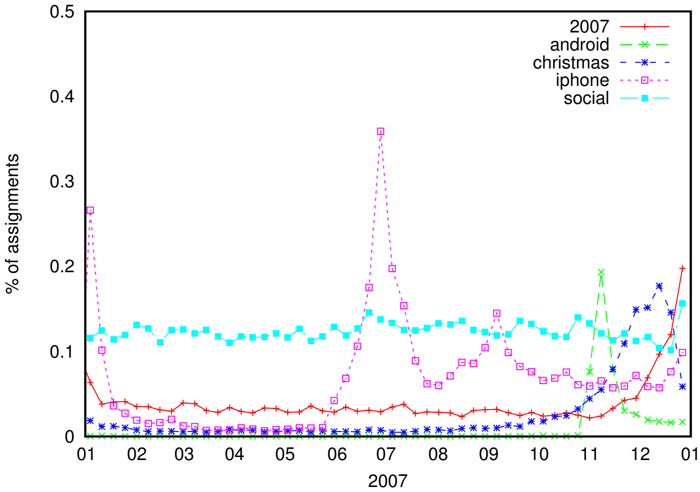
The diffusion-of-attention concept

In some cases, detecting spam may be computationally expensive or too ambitious. We therefore propose a new concept called *diffusion-of-attention*. Applying this concept drastically reduces the influence of spam on the global tag distribution without the actual need of filtering. The *diffusion-of-attention* approach is not a spam detection method in the conventional meaning of the term. However, we list it here as alternative that requires no supervision and was shown to perform well in tag-based trend detection scenarios (see Chapter 9).

We define the *attention* that a tag attracts in a certain period of time as the number of users who use the tag in this period. The rate of *diffusion* of a tag is then given by the number of users that assign this tag for the



(a) occurrence



(b) diffusion

Figure 4.3: Occurrence versus *diffusion-of-attention* for 5 sample Delicious tags in 2007. Figure (a) plots the relative occurrence of each tag in the *delicious* dataset over time. Figure (b) shows the number of users who assign the tag for the first time (*diffusion-of-attention*) as percentage of the overall first time assignments. The *diffusion-of-attention* concept captures all major trends also found in (a) except the peaking of the *social* curve in summer 2007. The spam activity that caused this peak is suppressed if moving to the *diffusion-of-attention* concept.

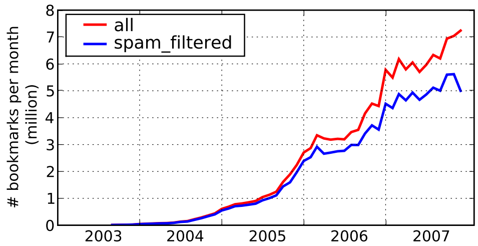
first time. This measures the importance of an item by its ability to attract new users. Every user's influence is thus limited, and a trend can only be created by groups of users. Figure 4.3 plots five example tag trends over time and compare the *diffusion-of-attention* measure to a count-based measure. The plots demonstrate that the *diffusion-of-attention* concept captures all major trends of count-based measurements. The measure successfully reflects seasonal trends, e.g. "christmas", as well as new tags rising in popularity. The only *trend* missing in the second plot is the peak of the "social" tag in June 2007. Further investigations show this peak to be caused by the activity of a single user who posted $5,666$ bookmarks all tagged as "social" within the relevant week. All these bookmarks link to the same domain. This spam does not influence our *diffusion-of-attention* measure, since it weights all users equally.

Moving to a diffusion perspective also yields efficiency improvements. Retaining only combinations of users and tags that appear for the first time in the Delicious corpus reduces the number of total tag assignments from 420 million to 102 million. We will successfully apply the *diffusion-of-attention* concept for the detection of trends in non-spam-free folksonomies in Chapter 9.

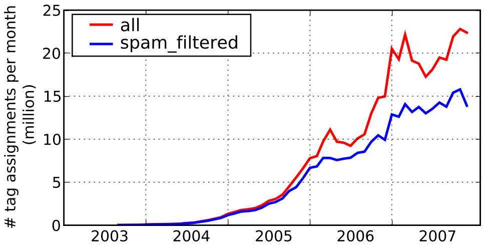
4.3 Cleaning Delicious

The removal of spam users from our *delicious* corpus required a fast and unsupervised detection method. We therefore decided to use a simple graph-based filtering method. The design of this filter was motivated by the observation that spammers tend to promote few domains with many posts (see also Figure 2.2). In order to identify spammers, our detector first counted the number of different URLs and domains per user. All users were then sorted by their domain to URL ratio. Assuming a spammer percentage of 10 percent, we removed those users with the lowest ratio of domains from the dataset. The threshold was chosen based on manual investigations which showed a spam ratio slightly below this value. Filtering the data in this way shrunk our corpus from 132 million to 109 million bookmarks. The resulting dataset is identical to the previously introduced $delicious_{clean}$ corpus.

Analyzing the effect of our spam filtering strategy on different aspects of the dataset, we observe that, though we only filtered 10 percent of all users, this lead to 28 percent decline in bookmarks and even 37 percent in tag assignments, as displayed by Figure 4.4. This further supports the intuition that spammers are highly productive and assign an above-average number of tags to promote their content. As a result, filtering spam reduces



(a) bookmarks per month



(b) TAS per month

Figure 4.4: The effect of spam filtering on the number of bookmarks and tag assignments per month for the Delicious dataset. Even though our spam filter only removed 10 percent of all users, we find the number of bookmarks to decrease by 28 percent and the number of tag assignments by 37 percent. This amplitude results from spammers' above-average bookmarking and tagging rates. The plots also show that spam is a recent phenomenon in Delicious, and the initial years of the service appear to be nearly spam-free.

the number of average tags per bookmark from 3.16 to 2.77, as reported in Section 2.3. Moreover, we find that spam is responsible for peaks in the number of tag assignments per month, which supports our observation of bulk uploads by spam users. Additionally, Figure 4.4 highlights that spam is a recent phenomena to Delicious, and the initial months of the corpus appear to be nearly spam-free.

Figure 4.5 illustrates that, though our spam detector does not explicitly consider any tag data, it filtered many of the users with an extreme high rate of tags per bookmark. On the other hand, our filtering method misses some users with an extreme tagging rate, but only few bookmarks. However, the impact of these spammers on the overall community activity is rather low.

4.4 Chapter conclusions

Spam drastically reduces the quality and credibility of recommender systems. It therefore needs to be filtered or suppressed before any sophisticated recommendation model can be built. This chapter presented an extensive study of the motivations behind folksonomy spam and the aspects that characterize the behavior of spam users. Among these characteristics we identified the clustering of spam users around only few domains as well as an abnormal tagging behavior due to a missing retrieval intention. Furthermore, this chapter gave a short overview of general approaches to spam detection, and described

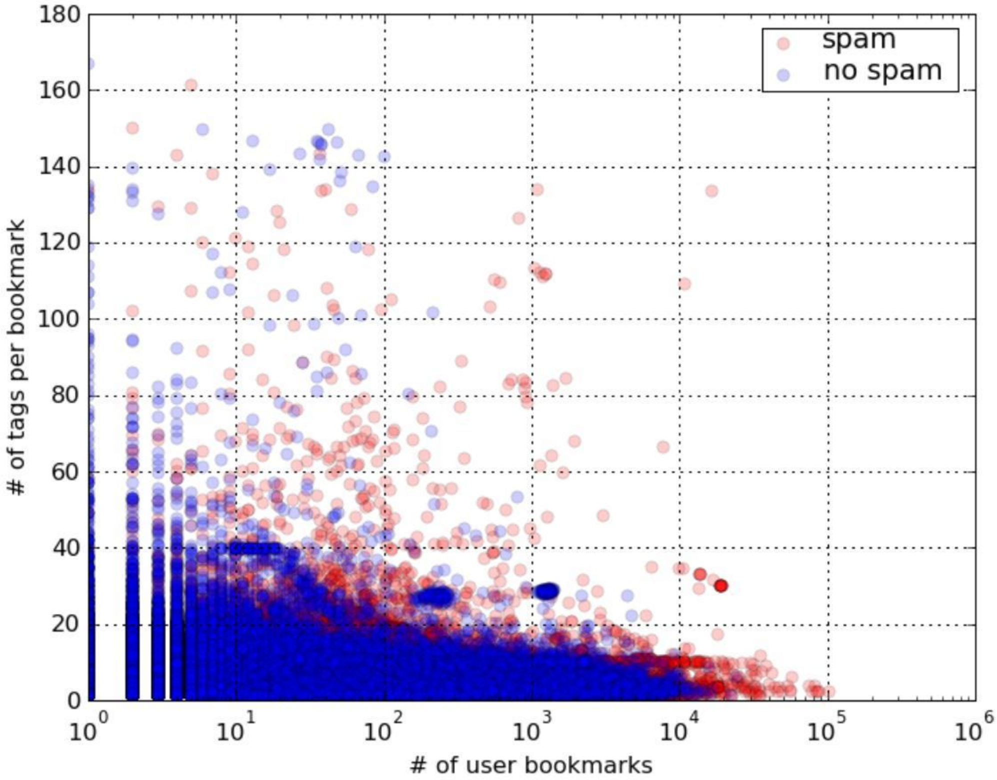


Figure 4.5: The number of bookmarks per user compared to the average number of tag assignments per bookmark for the *delicious* dataset. Users who have been identified as spam by our spam detector are in red. Most users with many bookmarks and an extreme number of tag assignments have been correctly classified as spammers, even though tags were not considered during the classification task. Other users with a very high tagging activity but fewer bookmarks have not been filtered correctly. However, the impact of these remaining spammers on the data is rather low.

the graph-based spam detector that was used to clean our *delicious* dataset. As noted in the previous chapters, we saw that the cleaning of our dataset affected a variety of dataset properties. A further contribution of this chapter is the introduction of the *diffusion-of-attention* concept for spam suppression. We will apply this concept to design more robust trend detectors in Chapter 9. Throughout Part II of this dissertation we will only consider folksonomy datasets that have been preprocessed in order to filter spam.

Chapter 5

Related work I

Folksonomies and collaborative tagging in general have attracted a growing interest in academia. The first discussions on the advantages and disadvantages of tagging date back to the year 2004 and took place in the blogosphere [Mat04]. The same year saw the coinage of the term "Folksonomy" in a blog post [Wal04]. All this happened only a few months after the launch of the Flickr photo sharing service in February 2004 and the Delicious social bookmarking service in September 2003. The exponential growth of these tagging services coincided with an exponential increase in related publications.

This chapter reviews publications that contributed to a better understanding of the concepts and dynamics behind collaborative tagging. Additionally, it presents previous work related to social spam detection. Other studies that focus on folksonomies with respect to potential recommendation applications will be extensively discussed in Chapter 10 instead.

5.1 Groundwork

The first discussions about tagging and folksonomies took place in the blogosphere. **Mathes** [Mat04] elaborates different aspects of collaborative tagging and compares tagging to well-known categorization tools, such as taxonomies. The author further investigates the reasons that led to the success of tagging and identifies low barriers of entry and reduced cognitive costs as core drivers. Moreover, he distinguishes between users that utilize collaborative tagging services for managing their own collections of content and users with a content sharing attitude. However, Mathes does not explicitly distinguish between broad and narrow folksonomies, and our own studies demonstrate that the activity in broad folksonomies is less motivated by content sharing than in narrow communities.

Thomas Vander Wal [Wal04] introduces the term "folksonomy" as syncrisis of the terms "folk" and "taxonomy". He later distinguishes between narrow folksonomies, where items are tagged by few users only, and broad folksonomies that support open tagging [Wal05]. **Shirky** [Shi05] juxtaposes ontologies and tagging and identifies environments that favor one of the both classification techniques. He argues that tagging is a superior classification technique in open, changing, non-expert environments. He further discovers power-law distributions in the number of tags per user as well as in the tag distributions of users and resources. Additional studies on the benefits and disadvantages of tag-based classification are presented by **Kroski** [Kro05] and **Quintarelli** [Qui05]. The discussion of the arguments for and against tagging presented in Section 1.1 of this thesis mainly summarizes the contributions of these authors.

Another highly influential early work on collaborative tagging is **Golder and Huberman** [GH06]. The authors present one of the first studies on the dynamics behind tagging and investigate the global and local growth of tag vocabularies within Delicious over time. Consistent with our own findings, they report that URLs in Delicious tend to be popular only for a few days. They also note that the tag distributions of URLs converge after few bookmarks. Their analysis is based on a small sample of $19,422$ Delicious bookmarks which renders the obtained results somewhat speculative and hard to generalize. However, our own analysis of a much more complete dataset supports most of their findings.

Marlow et al. [MNBD06] present a taxonomy for the categorization of tagging systems based on design choices, e.g. the existing tagging rights (who can tag what?) or the type of underlying resources (what can be tagged?). The authors further suggest to classify tagging systems according to the incentives of their users. These incentives may have an *organizational* or *social* character. In contrast to organizers, "social" users assign tags in order to share content, attract attention, or express their opinions. In accordance with our own findings, the authors further note that different incentives are likely to result in different tagging strategies. However, they do not investigate the relationship between differences in tagging rights and incentives.

Mika [Mik05] is the first to model folksonomies as tripartite hypergraphs, extending traditional bipartite ontology models by a user dimension. This representation has since then become the de facto standard, though different types of notation have been used. The author also proposes to reduce the hypergraph to three bipartite matrices $\mathbf{A}_{IT}$, $\mathbf{A}_{IU}$, and $\mathbf{A}_{TU}$ for easier handling, and to apply matrix operations in order to determine second order co-occurrences. All these concepts reappear throughout this thesis. Some recent publications aim to avoid the reduction to bipartite subgraphs, since it is generally accompanied by loss of information. Instead, they propose to directly work on the full folksonomy tensor [WZY06, SNM08, RBMNST09]. We will develop tensor-based models for the translation of tags in Chapter 8.

5.2 Tag classification

Various authors have studied the motivations behind tag assignments. The main contribution of the studies by **Golder and Huberman** [GH06] is the schematization of tag functions into seven categories, as described in Section 2.3. Their studies show the majority of tags to describe the content or type of a resource, but provide no detailed quantification. Furthermore, the authors do not investigate user-level differences in tagging behavior, and, according to our own studies, underestimate the importance of tags for content organization.

Additional evaluations on the categorization of tag functions are presented by **Bischoff et al.** [BFNP08]. The authors classify tag functions into eight categories similar to the ones identified in [GH06]. They then measure the distribution of tag types within different folksonomies, such as Delicious, Last.fm, and Flickr, finding different mixtures of tag functions for each service. Their results for Delicious mainly confirm our own observations, namely that the majority of Delicious tags describes the nature or topic of an item. However, for Flickr they report that around 25 percent of all tags refer to locations, whereas the majority of Last.fm tags refers to the genre of a song. Tag functions thus vary among domains and content types. The authors do not explore the correlation between different tag types, nor do they investigate user-level discrepancies.

Another tag categorization scheme is presented by **Xu et al.** [XFMS06]. The authors distinguish between content, context, and attribute tags, as well as subjective and organizational tags. They propose to exclude certain tag types, e.g. subjective tags, during the training of tag recommenders, since these are unlikely to be shared. However, they do not evaluate the effect of such filtering strategies. Furthermore, their tag recommender does not perform any personalization and thus does not have to reflect individual user needs.

Finally, **Sen et al.** [SLR+06] categorize tags into factual, subjective, and personal tags. They report that 63 percent of all MovieLens tags identify facts about a movie, compared to 29 percent subjective and only 3 percent personal tags. Further user studies revealed that the usefulness of the tags

	Bischoff [BFNP08]	**Golder** [GH06]	**Xu** [XFMS06]	**Sen** [SLR$^+$06]
1	Topic	About what or whom?	Content-based	Factual
2	Time	Refining categories	Context-based	
3	Location			
4	Type	What it is?	Attribute	
5	Author/Owner	Who owns it?		
6	Opinions/Quality	Qualities/Characteristics	Subjective	Subjective
7	Usage context	Task organization	Organizational	Personal
8	Self reference	Self reference		

Table 5.1: Tag classification schemes proposed by diverse authors and mappings [BFNP08].

from each class strongly depends on the user task. Motivated by this observation, the authors propose to design tag interfaces, where users can express their opinions separately from facts. The authors further note that users of MovieLens express considerably low interest in tagging, since many movies can be found directly based on title or actor information.

An overview of the discussed tag classification schemes gives Figure 5.1 [BFNP08].

5.3 Folksonomies & IR

Folksonomies have also been studied for their potential with respect to information retrieval. The most complete analysis of the Delicious bookmarking system next to our own work is provided by **Heymann et al.** [HKGM08]. The authors investigate the potential of collaborative bookmarking for web search. Among other results, the authors show that there exists a reasonably high overlap between search query terms and tags found in Delicious. Furthermore, they report that social bookmarking systems reflect changes within the underlying web structure, such as newly appearing or recently modified web pages, earlier than search engines or web directory services, such as the Open Directory Project. They also find that only 50 percent of all Delicious tags appear in the content of a web page. The authors conclude that social bookmarking services can provide additional information about items not available in content-based settings.

The distributional similarity between query terms and tags is also reported by **Carman et al.** [CBGC09]. The authors further propose the use of tags for smoothing document content models in order to improve document retrieval performance. **Krause et al.** [KHS08b] compare the distributions of Delicious tags and search terms and discover power-laws in both cases. They also find that tag-based item rankings and query results show high correlations in some settings. This correlation is especially high for topics related to information technology, which are well covered by Delicious. This lets the authors assume that Delicious users rely on search engines to find interesting content, which in turn is reflected in the bookmarked resources. These evaluations are complemented by the work of **Bischof et al.** [BFNP08]. In accordance with our own observations, the authors point out that some tags are more valuable for search than other, more personal tags. However, Bischof et al. assume that this regards only a small percentage of overall tags. Instead, our own studies show that the majority of tags serves individual tasks. This heterogeneity may be ignored in cases where cumulative effects create stable item vocabularies. However, these peculiarities are of crucial importance for user-centric retrieval scenarios. Furthermore, we believe that the information about who assigned which tag will help to design better retrieval models.

We will present a more detailed discussion of the related work on using folksonomies for information retrieval in Chapter 10.

5.4 Global network properties

Different publications have investigated network aspects of folksonomies, with the most exhaustive study probably being **Cattuto et al.** [CSB+07]. The authors adapt classical network measures like the *characteristic path length* or the *clusterness* of a network to a folksonomy setting and study the behavior of these properties over time. They show that folksonomies exhibit *small world* characteristics, since the *characteristic path length* tends to increase less then logarithmically with growing item, tag, and user sets. The authors report even constant path lengths of 3.5 hops in the case of Delicious . They therefore argue that the observed short path lengths facilitate the "serendipitous discovery" of interesting objects. During their analysis, the authors noticed a distortion of their data by spam, causing unexpected network behaviors. They therefore propose to design spam detectors that base their decisions on abnormal local network properties.

The power-law compliance of global item and tag node degrees has been known since the beginning of folksonomy research, e.g. **Hotho et al.** [HJSS06a] or **Heymann et al.** [Hey06]. Further evidence of this compliance in Delicious and other folksonomies was recently presented by **Krause et al.** [KHS08b], **Oldenburg et al.** [OGC08], as well as **Li et al.** [LGZ08]. Local power-laws of tag distributions have previously been reported for personomies by **Cattuto** [CLP07] as well as **Halpin et al.** [HRS07], for items by **Zanardi and Capra** [ZC08], and for co-occuring tags by **Cattuto et al.** [CLP07]. Whereas our own studies emphasize the power-law conformity of personomies, they also contrast [ZC08] by showing that items are dominated by very few tags way beyond an expected power-law inequality. This dominance is characterisitc for all item vocabularies. Power-laws are a common phenomenon in (social) networks and have also been found in a variety of other domains, including the word frequency distributions of natural languages, e.g. **Zipf** [Zip49], or the link structure of the World Wide Web, e.g. **Kleinberg** [Kle99]. An overview of power-law conform networks is given by **Chakrabarti and Faloutsos** [CF06].

The parabolic behavior of the user node degree distributions in folksonomies, which we encountered in Section 3.1, was previously reported by **Krause et al.** [KHS08a]. The authors further discovered similar patterns in the usage distributions of search engines. Other log-normal distributions have been found for citation networks by **Leskovec et al.** [LAH07], clickstreams by **Chakrabarti and Faloutsos** [CF06], as well as words in the bible by **Bi et al.** [BFK01]. **Bi et al.** also introduce the DGX distribution, a discretized version of the log-normal distribution. This distribution would be the natural choice for modeling the distribution of user node degrees. For a general overview on structural, functional, and dynamic properties of (social) networks see **Newman** [New03], **Chakrabarti and Faloutsos** [CF06], or **Leskovec et al.** [LAH07].

Information theoretic aspects of folksonomies have been analyzed by **Chi and Mytkowicz** [CM08]. The authors look at the evolution of the entropy of node degree distributions within Delicious over time. Consistent with our results, they observe a constant increase in entropy for items and a rather stable entropy for tags. **Halpin et al.** [HRS07] report convergence patterns within tag distributions over time. Calculating the Kullback-Leibler divergence between consecutive tag distributions, they confirm our observation that item vocabularies converge after an initial phase of randomness.

To the best of the author's knowledge, there exists no previous work that provides a comprehensive analysis about the coherence between the network properties of a folksonomy and their impact on potential recommendation scenarios.

5.5 Spam detection

The effect of spam on the quality of folksonomy data has been mainly ignored in past research. A recent study on the detection and prevention of spam in tagging systems is **Krause et al.** [KHS08a]. The authors describe their preliminary results on the construction of a spam detection framework for the Bibsonomy bookmarking system. They test standard machine learning approaches, such as Naïve Bayes classifiers or Support Vector Machines [MRS09], in order to identify the features that characterize social spam. Among the considered features are a user's co-occurrence with other users in the tag and item dimensions, profile data about a user, as well as location data derived from a user's IP address. The authors observe that co-occurrence patterns in the item space provide the most discriminative information. This means that new users can be identified as spammers if they bookmark the same URLs as previously identified spam users. The authors report that best detection accuracy for classifiers that consider the evidence of all features in their decisions.

Neubauer and Obermayer [NO09] discover that spam highly disturbs the component spectra of folksonomy graphs[1]. Analyzing the $bibsonomy_{spam}$ dataset, they report that nearly all next-largest components within the user-item graph G_{IU} are made up entirely of spam. These findings support two of our own observations: (1) spammers tend to bookmark resources of little interest to other users, and (2) spammers use different accounts in order to boost the popularity of their content. Based on this evidence, Neubauer and Obermayer [NO09] propose an unsupervised spam detector that classifies users as spam if they fall into next-largest components. This approach achieved reasonably good detection results for spammers with large collections of bookmarks, but performed less satisfactory in *cold-start* situations. Extending their examinations to tags, the authors find that the personomies of users and spammers overlap, such that tags have less discriminative power than items. Nevertheless, the best detection accuracy was achieved when combining tag and usage-based evidence.

Further publications on social spam detection come from the participants of the 2008 Spam Detection Challenge that was hosted during the "European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases" (ECML PKDD). Participants had to develop spam detectors for the Bibsonomy bookmarking service which were evaluated on the manually labeled $bibsonomy_{spam}$ dataset. **Gkanogiannis and Kalamboukis** [GK08], the winning team, reached a detection accuracy of above 97 percent by applying well-known text classification techniques. Their method detects spammers based on the tags they assigned or key terms within the posted URLs. A promising direction for further improvements is the combi-

[1] Components are those subgraphs of a graph that are not connected to each other. The giant component is the largest of these components. All other components that are not isolated, i.e. they consist of more then one node, are called next-largest components. [NO09]

nation of text-level detectors and graph-based methods [NO08].

Koutrika et al. [KEG+07] do not consider the problem of spam detection, but try to reduce the influence of spammers in a tag-based search scenario. To this end, they weight each user by her reliability given as the number of times another user tagged an item of this users collection with the same tag. Hence, a user who has a bookmarking behavior similar to the overall community will receive a high weight. The popularity of each item i with respect to a given tag t is then calculated as the weighted sum of all users that previously labeled i with t. We believe that the basic assumption of Koutrika et al., namely that users are reliable if there exist other users with similar bookmarking behavior, is misleading. Instead, our own evaluations show that some spammers use different user accounts to promote the same items with the same tags. These different accounts will all contribute to each others credibility, leading to high reliability values for a certain type of spammers. This might even happen if all these accounts are not connected to the giant component. A more robust spam suppression measure may instead consider a user's connectivity with respect to the giant component, as proposed by Neubauer and Obermayer [NO09].

Yeung et al. [AYGS09] as well as **Xu et al.** [XFMS06] also consider the problem of spam suppression. Both research teams use a weight diffusion method similar to the HITS algorithm [Kle99] in order to derive authority scores for users. Yeung et al. define authority based on a user's level of connectivity in the (weighted) item space, whereas Xu et al. assume authority if a user exhibits a similar tagging behavior as the overall community. Yeung et al. show that their method ranks spammers lower than non-spammers in a tag-based expert finding scenario. We will present a more detailed discussion of both methods in the context of existing recommendation approaches in Chapter 10.

5.6 Datasets

The final section of this chapter provides an overview of previous publications from a data perspective. Research on folksonomies and collaborative tagging centers around a small set of real-world services. Probably the best researched folksonomy to date is **Delicious**. This comes to no surprise, since Delicious was one of the first popular social bookmarking services on the web. The pure size of the Delicious service in the number of users and bookmarked items allows studies that cannot be performed on other services. Moreover, Delicious is probably the "purest" real-world folksonomy, since its structure closely resembles the tripartite folksonomy graph of items, tags, and users, without offering much additional functionality. This allows the study of the dynamics of collaborative tagging without much interference by other factors. Consequently, most early discussion and literature on broad folksonomies is on, or references, the Delicious service, see e.g. [Mat04, Mik05, Shi05, Cat06, GH06]. The most comprehensive Delicious study, that investigates the potential of social bookmarking services for enhanced web search, is [HKGM08]. The study is based on a corpus of about 22 million Delicious bookmarks. Existing literature on Delicious is based on proprietary datasets, making the comparison of the presented results cumbersome. Furthermore, every dataset was crawled in a different way, such that it is often unclear whether new findings stem from biases in the crawling methods or not. We hope that the release of our Delicious dataset will help to change this situation.

Another popular research object is the **CiteULike** folksonomy which supports the bookmarking of publications. Every user can annotate publications, bibliographic meta-data, e.g. by adding information about the authors, title, or publisher. Despite the availability of the data, few authors have actually looked at the patterns that emerge within the CiteULike service, e.g. [SNCA+09, OGC08]. Instead, most literature uses CiteULike for the evaluation of novel retrieval solutions [ZC08, BvdB08, Hey06, SWUH09]. This tendency is probably due to the rather small size of the CiteULike dataset when compared with other services, like Delicious or Flickr.

Similar in size are the datasets that are constantly made available by the **Bibsonomy** team. The Bibsonomy service allows to bookmark URLs together with publications, unifying the intentions behind Delicious and CiteULike. Bibsonomy is the only dataset that has been manually controlled for spam. In consequence, we find much literature on spam detection in folksonomies based on, for instance, the $bibsonomy_{spam}$ dataset [NO08, GK08, KHS08a]. Furthermore, Bibsonomy has become a popular benchmark dataset for the evaluation of tag recommenders, since it was used during the 2008 and 2009 ECML PKDD discovery challenges on tag recommendation [MBA+09, RST09, KF09, WSZ09] as well as in influential early publications about the same topic [JMH+07]. The relatively small size of the current Bibsonomy datasets impedes comprehensive studies on network properties, though some work exists [CSB+07].

Yet another popular folksonomy service is **Last.fm**, where users assign tags to music. As for Delicious, there exists no commonly shared Last.fm dataset. The most comprehensive analysis of the Last.fm folksonomy is probably [BFNP08] that examines the tagging behavior of users in different folksonomies. Other work on Last.fm is mainly application-oriented [RBMNST09, TSMT08, JMH+07].

Folksonomy	Publications
Delicious	[Mik05, Cat06, GH06, Hey06, HJSS06a, HJSS06b, Vos06, WZY06, BXW+07, CSB+07, CBSL07, GHS07, HRS07, JMH+07, NM07, YJNT07, ALMP08, CBC08, CBHS08, CM08, DS08, HKGM08, HRGM08, KHS08b, LGZ08, OGC08, SGMB08, SZL+08, Str08, SVG08, SCA08, WZB08, WPK+08, YBLS08, mAYGS08, GBM+09, NO09, NAYG+09, RHMGM09, WZB10a, WUS09, AYGS09, YLMH09, WZB10b]
Bibsonomy	[HJSS06a, CSB+07, JMH+07, CBC08, KHS08a, Lip08, NO08, OGC08, SZG08, JEHS09, LYHM09, MCM+09, MBA+09, NO09, RBMNST09, WSZ09, ZML09, WZB10b]
CiteULike	[Hey06, BvdB08, OGC08, SZL+08, ZC08, NO09, SWUH09, SNCA+09]
Last.fm	[JMH+07, BFNP08, SGMB08, TSMT08, RBMNST09]
MovieLens10M	[SLR+06, SHLR07, SVR09]
Flickr	[DKM+06, MNBD06, AN07, SCA+07, AS08, BFNP08, GW08, SvZ08, WSVZ08, vZMGPR08, CMG09, LHY+09, NPY+09, OSvZ09, SS09]

Table 5.2: Services commonly analyzed in folksonomy research and corresponding publications that appear throughout this work.

A newcomer among the folksonomy datasets is the *movielens10M* corpus. The dataset provides the ratings and tags users have assigned to movies. Together with the Netflix corpus[2], the **MovieLens** datasets are highly popular for research on rating prediction as well as movie recommendation. However, the ratio of tagged movies in this data is still very low (below 0.25 percent), reducing the value of the dataset for folksonomy research. As a result, existing work that evaluates the contributions of tags in this domain mainly originates from the providers of the dataset [SHLR07, SVR09].

The best understood narrow folksonomy is probably **Flickr** which has influenced folksonomy research since its beginnings [Mat04, Shi05, MNBD06]. The first influential study of tag usage and vocabulary formation in Flickr is Marlow et al. [MNBD06] which was later extended by Bischoff et al. [BFNP08]. Most of the recent publications on Flickr are composed by members of the Yahoo! research team[3]. Their studies mainly focus on the incorporation of tags for enhanced photo retrieval [vZMGPR08, OSvZ09,

[2] `http://www.netflixprize.com/`

[3] Yahoo! acquired Flickr in 2005 [Fli05].

WSVZ08]. We will discuss these works in more detail in Chapter 10. Table 5.2 lists the most popular folksonomies in terms of research coverage and relates publications that appear throughout this thesis.

The successful design of recommender systems requires a deep understanding of the underlying data. Part I of this dissertation therefore was dedicated to the exploration of the temporal and structural patterns that emerge in folksonomies. Building on these analytical findings, the second part now presents novel recommendation algorithms that have been especially designed to capture the nature of folksonomies.

Part II

Recommendation in Folksonomies

Chapter 6

Introduction

Recommender systems have become an integral part of most web portals. They help people discover interesting books on Amazon[1] and LibraryThing[2], promising movies on IMDB[3] and Netflix[4], videos on YouTube[5], websites on StumbleUpon and Google[6], news on Digg[7] and Google news[8], music on Last.fm and Apple's iTunes[9], or potential partners on Parship[10]. Other types of recommenders support users during common tasks by recommending prospective next actions, related content, tags, or categories. Recommender systems help users to cope with the ubiquitous information overload by presenting only a selection of potentially useful alternatives out of an overwhelming set of choices. This results in better usability and increasing service profits due to a more efficient mapping of supply and demand. Furthermore, recommender systems can guide users towards interesting content that they had been unaware of and would not have actively searched for.

Folksonmies can benefit from recommendation technology in many areas. A common feature of existing folksonomies is tag recommendation: the suggestion of relevant labels. In addition to its positive effect on usability, this feature assists in lowering the rates of misspellings and to increase the rate of tag reassignments with positive effects on global and local tag convergences and thus on potential search scenarios. An even higher benefit arises in item

[1] http://www.amazon.com
[2] http://www.librarything.org
[3] http://www.imdb.com
[4] http://www.netflix.com
[5] http://www.youtube.com
[6] http://www.google.com/history/items
[7] http://digg.com
[8] http://news.google.com/
[9] http://www.apple.com/itunes/
[10] http://www.parship.com/

recommendation scenarios, where users are supported in the discovery of content that satisfies their information needs. Such scenarios are especially appealing, since folksonomies aggregate the interests of potentially millions of users. Furthermore, the manual labeling of content by tags produces high-quality descriptors regardless of content heterogeneity. These tags can be used to index and compare content without the actual need of expensive content collection and feature extraction, making folksonomy recommenders a promising alternative or complement to existing search technologies.

From an information retrieval perspective, recommendation can be seen as the problem of filtering and ranking a given set of objects with respect to the present context. This context may manifest in various forms. For example, many scenarios will require the personalization of recommendations for a given user with individual information needs. Additional context may be embodied by a user's current intention which can be given explicitly, e.g. in the from of a search query, or implicitly, e.g. inferred from previous user activity. The identification of relevant context information and its incorporation into the recommendation logic is a major challenge in the design of successful recommender systems. Since this work only considers the problem of graph-based recommendation, context can occur solely in the form of one or more objects from the item, tag, and user sets. For instance, the problem of personalized item recommendation resembles the task of item recommendation in the context of a given user. We introduce and categorize the recommendation scenarios that materialize from different context definitions in Section 6.2. From these potential scenarios, this thesis will focus on the most relevant settings and present novel solutions to the challenges of personalized item recommendation (Chapter 7), (personalized) social search (Chapter 8), and tag recommendation (Chapter 8).

Recommender systems are historically categorized into content-based, collaborative filtering, and hybrid methods [AT05]. Standard collaborative filtering techniques infer a user's preferences based on the activity of other users and are generally also applicable to folksonomies. However, a simple transfer of existing collaborative solutions to folksonomies disregards the additional information contained in tags. Furthermore, those approaches will likely suffer from data sparsity caused by the open nature of folksonomies (see Section 3). We will show that the incorporation of tags into the recommendation model helps to reduce this sparsity, resulting in more descriptive and robust user and item models. Moreover, we find that recommenders that derive item similarities based on tags perform reasonably well compared to usage-based recommenders. However, tag-based approaches can be improved by combining usage and tagging-based observations into a unified model, as we will show in Chapter 7. The vast majority of previously proposed solutions to the problem of folksonomy recommendation reduces the folksonomy hypergraph to a set of bipartite subgraphs before building their models. This simplifies the design of recommenders, since it allows the reuse of well-known techniques from other bipartite scenarios, such as movie recommendation or document retrieval. Unfortunately, this simplification is traded for a loss of information, leading to suboptimal recommendation results. We therefore propose a tensor-based recommendation approach in Chapter 8 which is especially tailored to the tripartite hypergraph nature of folksonomies.

Folksonomies are highly dynamic communities, and the relevance of recommendations may drastically decay over time. Another form of context is therefore given by the point in time a recommendation is made. We will touch on the temporal aspects of recommendation by looking at the task of tag-based trend detection in Chapter 9. Trend detection may help the users of a folksonomy to discover upcoming topics at an early stage. We will present a novel probabilistic generative model for early trend detection and compare its performance to previously presented methods.

This chapter is intended to give an overview of the problem of recommendation in general, and folksonomy recommendation in particular. It describes the principal concepts behind existing recommendation techniques and introduces evaluation measures commonly used in a recommendation context. We then discuss the task of recommendation from a folksonomy perspective and provide a schematization of the potential application areas for graph-based folksonomy recommenders. The chapter concludes with a brief overview of general strategies to folksonomy recommendation and an accentuation of the contributions of this thesis.

6.1 The recommendation task

The task of recommendation can be formulated as the problem of selecting N objects from a known set O that maximize a utility function g with respect to a given context c. In the case of folksonomies these objects may refer to items, tags, or users, depending on the type of object that has to be recommended. Context depends on the recommendation setup and may represent a user, e.g. in personalized item recommendation, or a user-item combination in tag recommendation. The recommendation task can hence be formulated as

$$\widehat{O}(c) = \arg \overset{N}{\max}\, g(c, o),$$

where the utility function $g(c, o)$ measures the usefulness of object $o \in O$ in a given context c. The challenge of recommendation is the estimation of g based on incomplete information [AT05].

In many recommendation environments, e.g. in movie recommendation, the utility of an object for a user is represented by a rating. However, the folksonomies we consider in this work do not provide any rating information. Instead, we have to assume that a user bookmarks an item only if she likes it. Further, we need to suppose that she only assigns tags which she herself considers useful. The recommendation task in folksonomies therefore resembles a link prediction problem, where we want to predict $g(c, o) = 1$ if a future version of the folksonomy graph G contains an edge between the nodes from the context and o, and $g(c, o) = 0$ otherwise. For instance, in a personalized item recommendation scenario we want to predict items that a user would bookmark, but is not yet aware of. Since we only consider graph-based scenarios, we can solely recommend objects and model contexts that have been known at the time of training. This means that recommendation and context objects have to be part of the folksonomy graph at training time. These limitations do not apply for other recommender types, e.g. content-based recommenders, which can derive recommendations for new context instances by looking at object related attributes or an object's content.

The direct estimation of g proves to be rather complex, if not impossible. Instead, it has become common practice to develop ranking-based approaches, where only the order of the values in g needs to be estimated. The recommendation problem then becomes

$$\widehat{O}(c) = \arg \overset{N}{\min}\, r(c, o),$$

where $r(c, o)$ is the ranking function. The task of estimating g is hence transformed to the task of estimating a ranking function r based on incomplete information.

The next section briefly introduces the general design principles of recommender systems. These principles will later reappear in various forms when we will discuss existing recommendation strategies in a folksonomy environment.

General approaches

According to [AT05], recommender systems can be classified into one of three categories: content-based recommenders, collaborative filtering recommenders, and hybrid approaches.

Content-based methods

Content-based methods estimate the utility function $g(c, o)$ by calculating the *similarity* between the content of an object o and (the content of) those objects that fall into c. There exist many possibilities to derive object similarities based on content. For instance, a common way to measure the similarity between two text documents is to model each document as a word vector or word distribution before calculating geometric or probabilistic distance measures[11]. Content-based methods suffer from a variety of limitations [AT05, BS97]. Objects may come with little content information, or content features may be difficult to extract, e.g. in the case of videos or music. Moreover, objects may be heterogeneous in their types, formats, or languages, making their modeling expensive and limiting their comparability. Finally, as noted by [AT05], content-based methods fail in cases where no or little context history is given. This can occur in cold-start situations, where objects have to be recommended to new users who have not yet shown any preference for other objects.

Collaborative filtering

Collaborative filtering, unlike content-based methods, predicts the utility of an object based on the past activity of other users. Objects are assumed to be similar if they have been bought, watched, or equally rated by the same users. On the other hand, similar user interests can be inferred in cases where two users tend to buy, watch, or equally rate the same or similar objects. Thus, objects can be recommended to a user based on the preferences of similar users (*peers*) or by identifying objects that have attracted the same users as the objects the user liked in the past. These approaches are generally referred to as user-centric and item-centric collaborative filtering [HKTR04].

Collaborative filtering approaches are content independent and therefore do not suffer from most of the limitations inherent to content-based methods. However, collaborative methods fall short when recommending long tail objects, where little usage information is known. Furthermore, as for the content-based approach, collaborative filtering requires a certain amount of historical user data in order to build significant models. This limits its applicability when confronted with new users. Another classical problem of collaborative recommenders is data sparsity. Information about each user's preferences will only be available for a very small fraction of the overall object set, leading to poor recommendation results, especially for users with unusual tastes. Various solutions try to reduce this sparsity by performing densification techniques, such as clustering [Hof99b, GSRM09], matrix factorization [Kor09] or by using stereotypes [SSH97]; that group users by predefined criteria.

[11] See [MRS09] for an overview of document modeling and comparison techniques.

Hybrid methods

Hybrid methods combine content-based and collaborative methods to limit the shortcomings of each of the separate solutions. Hybrid recommenders can be built in different ways. One possibility is to combine the recommendations by an ensemble of recommenders in a final processing step. Other approaches enrich existing models by incorporating observations from other sources [SKKR01, MMN02], e.g. by additionally calculating object similarities based on meta-data. Finally, hybrid recommenders can incorporate information from multiple sources into a unified model, e.g. by clustering documents based on content and usage patterns in parallel [SPUP02]. We will discuss the applicability of these standard approaches to the problem of graph-based folksonomy recommendation in Section 6.2. For a more complete overview of recommender systems see [AT05] and [HKTR04].

Evaluation measures

We will evaluate our recommendation methods based on a variety of standard evaluation measures that are presented below.

Precision@N

The precision of a recommended object set $\widehat{O}_c$ with $|\widehat{O}_c| = N$ given a set O_c of objects that are relevant in a given context c is defined as

$$Prec@N = \frac{|\widehat{O}_c \cap O_c|}{N}.$$

The precision is thus a measure of the quality of the recommended objects.

Recall@N

The recall achieved by a recommended object set $\widehat{O}_c$ with $|\widehat{O}_c| = N$ is defined as

$$Rec@N = \frac{|\widehat{O}_c \cap O_c|}{|O_c|}.$$

The recall thus measures the completeness of the recommendations.

$\mathbf{F_1}$-measure@N.

It is generally possible to reach maximal recall rates by recommending a vast amount of items. However, this will lead to a drop in precision. The

F-measure combines recall and precision into one measure and is defined as

$$F_1@N = \frac{2 \cdot Prec@N \cdot Rec@N}{Prec@N + Rec@N}.$$

The subscript of 1 expresses the fact that precision and recall are weighted equally.

Maximal $\mathbf{F_1}$-measure

So far, all introduced evaluation measures were rank sensitive. These measures may suffice in scenarios that require a fixed number of recommended objects, e.g. due to interface requirements. However, in many other scenarios it may be useful to measure the quality of the entire list of recommendations. One measure that reduces the quality of a result list to a single number is the **maximal $\mathbf{F_1}$-measure** which is the F_1-measure that is achieved when recommending an optimal number of objects:

$$F_1^* = \max^{N} F_1@N.$$

Average Precision and Mean Average Precision (MAP)

Another very popular quality measure is Mean Average Precision (MAP) [MRS09]. The **Average Precision** for a single recommendation is the average of the precision values obtained at the ranks of the objects in O_c. The Average Precision of a ranking can thus be calculated as

$$AP = \frac{1}{|O_c|} \sum_{o \in O_c} Prec@r(c, o).$$

The **Mean Average Precision** is then defined as the mean of the Average Precision values over all contexts C' of the test set:

$$MAP = \frac{1}{|C'|} \sum_{c \in C'} \frac{1}{|O_c|} \sum_{o \in O_c} Prec@r(c, o).$$

ROC curves and AUC value

Receiver operating characteristics (**ROC**) graphs are 2-dimensional graphs in which the ratio of true positive recommendations at each rank N is plotted on the y-axis, whereas the x-axis displays the rate of false positives [Faw06]. Each point of the curve is thus defined as

$$ROC_{x,y}@N = \left(\frac{N - |\widehat{O}_c \cap O_c|}{|O| - |O_c|}, \frac{|\widehat{O}_c \cap O_c|}{|O_c|} \right).$$

A perfect recommender will rank positive objects at the beginning of the result list. The corresponding ROC curve will therefore grow in the y-direction until it reaches its maximal value of one. The curve will then continue in the x-direction till N is maximal and x is one, once the remaining list solely contains negative examples. The area under such a curve (**AUC**) will be maximal and equal to one, whereas a worst case recommender that recommends positive items in the end will have an AUC value of zero. The AUC value thus reduces the ROC curve to a scalar quality measure:

$$AUC = 1 - \frac{1}{|O_c|(|O| - |O_c|)} \sum_{i=1}^{|O_c|} r(c, O_{c,i}) - i.$$

The AUC of a recommender is equivalent to the probability that the recommender will rank a randomly chosen positive instance higher than a randomly chosen negative instance [Faw06]. The AUC value of a random ranker is therefore expected to be 0.5.

For an extensive discussion of ROC curves and the AUC value see [Faw06]. All other information retrieval measures are described in [MRS09].

6.2 Recommendation in folksonomies

Recommendation in folksonomies has many facets. The most obvious scenario is the recommendation of items, since this corresponds to well-known document retrieval tasks. A further well-researched recommendation task is tag recommendation which resembles the classical problem of content classification, potentially extended by a personalization aspect. Finally, it is also possible to recommend other users, e.g. in an expert finding scenario. In the following subsections, we will discuss and categorize these different recommendation settings and present existing approaches that have been developed to tackle some of these settings.

Scenarios

One way to group the recommendation scenarios that emerge in a folksonomy context is to distinguish the types of objects that are recommended as well as the context in which these recommendations take place. Table 6.1 presents a matrix of possible recommendation scenarios and their most common use cases in practice. The tripartite nature of folksonomies results in six potential forms of context that correspond to each node type as well as all possible node type combinations.

Table 6.1: Recommendation scenarios in folksonomies. The input represents the context dimension a recommendation is based on, and the output denotes the object types that are recommended.

Input/Output	items	tags	users
item	Recommendation of similar items	Item classification	Item-aware expert finding
tag	Social search	Tag classification Translation	Topic-aware expert finding
user	**Personalized recommendations (Chapter 7)**	User classification	Personalized expert finding
item + tag	Recommendation of similar items by topic	Iterative tag recommendation	Topic- and item aware expert finding
item + user	Personalized recommendation of similar items	**Tag recommendation (Section 8.4)**	Personalized, item-aware expert finding
tag + user	**Personalized social search (Section 8.5)**	**Tag translation (Chapter 8)**	Personalized, topic-aware expert finding

The most extensively researched recommendation setting is **tag recommendation** [JMH+07, HRGM08, WSZ09]. The goal in this scenario is to correctly predict the tags a given user will assign to a given item. Figure 6.1 shows the tag recommendation interface of Delicious as it appears during the posting a new URL (http://slashdot.org). The user can select tags from two suggested sets. The second set consists of the most popular tags for the given URL, whereas the first set seems to be more related to the users previous tagging behavior, although details remain unknown.

Yet another challenging task is **item recommendation**, the prediction of folksonomy items that will be interesting for a given user. Figure 6.2 presents an example of the weekly email newsletter sent by the StumbleUpon bookmarking service. The newsletter informs users about recently popular URLs from a global and a personalized perspective. Even though the StumbleUpon bookmarking service cannot be considered a folksonomy, since tagging only plays a secondary role, similar recommendation scenarios can be envisioned for folksonomies, too. The item recommendation task reappears in classical social search scenarios, where a user searches for content by tags, as well as in more complex scenarios, such as personalized social search.

User recommendation scenarios, on the other hand, have attracted less interest among researchers, e.g. [NAYG+09]. This absence of studies prob-

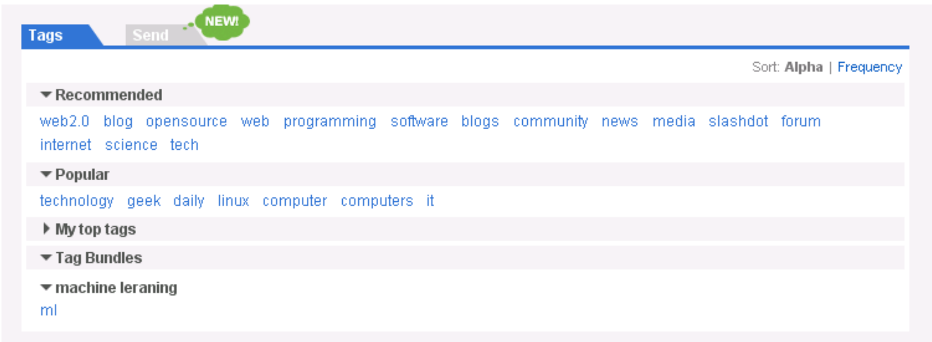


Figure 6.1: Tag recommendation interface of the Delicious bookmarking service (Sep 2009). The user can select from two sets of tags, one more personalized and one with the most popular URL tags.

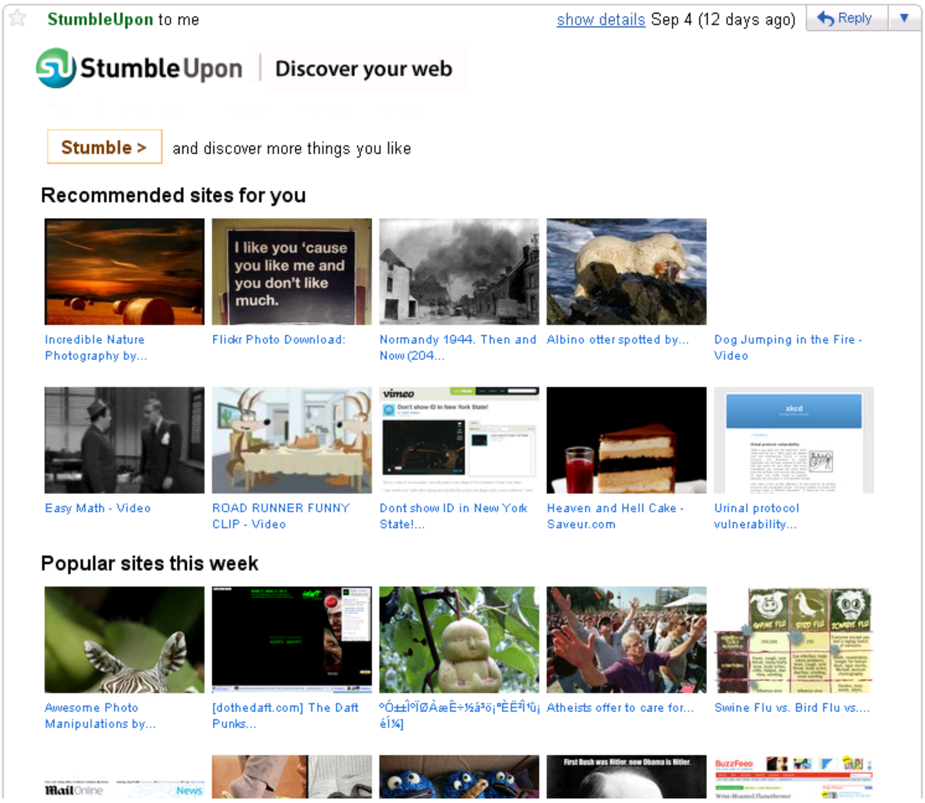


Figure 6.2: Example of an email newsletter from the StumbleUpon bookmarking service. The newsletter informs users about potentially interesting new websites (Sep 2009). Similar to the tagging interface of Figure 6.1, recommendations are grouped into personalized and popular items.

ably results from the lack of measures of user expertise which impedes the evaluation of the proposed approaches. For the same reasons, this thesis will concentrate on tag and item recommendation scenarios. In particular, we will present solutions to the problems of personalized item recommendation (Chapter 7), personalized social search (Section 8.5), and tag recommendation (Section 8.4).

All scenarios in Table 6.1 can be further divided based on the characteristics which influence the relevance of an object. One such characteristic is the novelty of an object, since this can strongly influence its relevance. Section 9 therefore discusses the problem of tag-based trend detection within folksonomies and presents a novel approach to this problem based on generative probabilistic models.

General approaches

The 3-uniform hypergraph nature of folksonomies distinguishes them from standard recommendation scenarios which consist of one or more 2-uniform hypergraphs. However, in order to apply standard recommendation techniques, it has become common practice to reduce the folksonomy graph to a set of 2-uniform hypergraphs. The most common reduction technique is the aggregation over one mode of the folksonomy tensor $\boldsymbol{\mathcal{Y}}_{ITU}$ which produces the matrices $\mathbf{A}_{IT}$, $\mathbf{A}_{IU}$, and $\mathbf{A}_{TU}$[12]. These matrices then act as input for common recommendation techniques. For instance, aggregating over the tag dimension of the $\boldsymbol{\mathcal{Y}}_{ITU}$ tensor produces the item per user matrix $\mathbf{A}_{IU}$, resembling a common collaborative filtering setting. However, at least for broad folksonomies the reduction to bipartite graphs always results in loss of information with a negative impact on recommendation quality. This led to the development of tensor-based methods that directly work on the $\boldsymbol{\mathcal{Y}}_{ITU}$ cube. We discuss existing approaches to folksonomy recommendation throughout this section.

Tag-based models

Since this dissertation only considers graph-based recommenders, we cannot model items by content features in order to apply content-based recommendation techniques. However, as discussed in Section 2.3, we find that the distributed tagging activity in folksonomies produces characteristic and stable descriptors for the underlying items. Each item can therefore be modeled by its characteristic tag distribution or by a tag vector. Tags then substitute words in classical text document modeling, and it is generally assumed

[12]See Figure 1.4 for details.

that the semantic quality of tags even exceeds the one of document terms [HKGM08]. Tag-based models can hence be expected to produce a recommendation quality at least equal to the one of classical document models. Furthermore, tags provide homogeneous descriptions for different content types and formats, allowing for the comparison of heterogeneous items.

One popular way to measure the similarity between folksonomy items is to calculate the cosine between their tag vectors [SGMB08, GBM+09]. The cosine between two objects is defined as

$$cos(o_1, o_2) = \frac{\mathbf{v}(o_1) \cdot \mathbf{v}(o_2)}{|\mathbf{v}(o_1)||\mathbf{v}(o_2)|},$$

where $\mathbf{v}$ denotes the vector representation of each object [MRS09]. For folksonomy items this will generally be a vector in the tag space which's values correspond to the number of times a tag was assign to an item. Other authors, such as [LGZ08] and [CBSL07], further stress the similarity to content-based document models and normalize tag distributions, e.g. by applying the well-known *tf-idf* normalization [MRS09]. These similarity calculations are often combined with clustering or rank reduction methods [SZL+08, LGZ08, KF09, ZWY06]. We will later evaluate the performance of a tag-based item recommender as special case of the hybrid model presented in Chapter 7.

Collaborative Filtering

A straightforward implementation of a collaborative filtering algorithm could determine the similarity between folksonomy items not based on tags, but on the users that bookmarked these items. Building the model of such a recommender solely requires to look at the matrix $\mathbf{A}_{IU}$ and apply standard collaborative filtering techniques. However, although such item-centric collaborative filtering models were shown to perform reasonably well in movie recommendation settings [SKKR01], they suffer from serious problems in a folksonomy environment. As discussed in Chapter 3, these problems are mainly a consequence of the open nature of folksonomies that leads to a much higher sparsity of the $\mathbf{A}_{IU}$ matrix. The problem of sparsity also impedes user-centric collaborative filtering solutions, where the similarity of users is derived from overlapping bookmarking patterns. This difficulty probably also explains the relative lack of existing collaborative filtering approaches that only work on the $\mathbf{A}_{IU}$ matrix. Instead, it seems much more popular to consider two users as similar if they exhibit similar tagging behaviors. For instance, the authors of [ZC08, SCA08, FNP07] all derive user similarities based on the cosine distance between the tag distributions of personomies.

While the comparison of users based on simple personomy coincidence seems straightforward, we find this basic procedure to strongly suffer from the heterogeneity in users' tagging behavior. As we will describe in Chapter 8, users of folksonomies develop their own vocabularies of tags, choosing from a pool of synonymous, singular plural, abbreviations, or multilingual tags. Furthermore, the basic method fails in cases of homonyms, where the same tag has multiple meanings. The direct comparison of tag vocabularies hence produces suboptimal user similarities. We will present a model that overcomes these problems in Chapter 8. Our method infers the meaning of each tag based on who used it and the items it was assigned to. This allows us to translate user tags to the global tag vocabulary before calculating any similarities.

Hybrid models

In classical recommender design, many authors try to combine content-based and collaborative models in order to avoid the shortcomings of each separate method. Analogously, we observe a tendency towards hybrid recommenders for folksonomies, with content-based models being replaced by tag-based ones. Hybrid folksonomy recommenders generally consider the information of two or all bipartite matrices $\mathbf{A}_{IU}$, $\mathbf{A}_{IT}$, and $\mathbf{A}_{TU}$ in parallel.

The most common hybrid scenario uses tags as intermediates between users and items [NM07, CBC08]. Items and users are consistently modeled in the tag space, such that item recommendations can be derived by identifying the item vocabularies that best match a user's personomy. Algorithmically, this is equivalent to the construction of the matrix $\mathbf{A}_{IU}$ as the product of the (normalized) matrices $\mathbf{A}_{IT}$ and $\mathbf{A}_{TU}$. This approach helps avoid common problems related to the sparsity of the original $\mathbf{A}_{IU}$ matrix. Variations of this method map users and items to a latent topic space first, before calculating user item distances within this latent space [XBF+08]. However, all these approaches disregard the aforementioned user-centricity of tagging. Users with more individual tagging behavior are therefore likely to get poor recommendations. Furthermore, as reported in Chapter 3, we find different distributional patterns for personomy and item tags, such that their direct comparison is problematic.

A further type of hybrid methods tries to enhance classical collaborative filtering methods by incorporating tags. The similarity between items or users is then not only determined from usage patterns, but also by the corresponding similarity in the tag space. Generally, there exist two strategies of tag incorporation. Models can be trained independently and the recommendations by each model are then merged in a post-processing step.

For instance, the authors of [XBF+08] follow this strategy in a social search scenario and derive the final item rankings from a mixture of usage and tag-based rankings. Alternatively, it is possible to train a unified model which already incorporates all sources during training. We will present a probabilistic hybrid recommender which develops a unified latent model from the observations contained in $\mathbf{A}_{IU}$ and $\mathbf{A}_{IT}$ in Chapter 7. Our evaluations show that the incorporation of tags helps drastically improve the quality of recommended items if compared to basic collaborative filtering approaches.

A common scenario that requires hybrid solutions is tag recommendation. Here, recommendations have to consider the given item and user input in parallel. Successful tag recommenders therefore have to take into account the tags that are common for an item, represented by $\mathbf{A}_{IT}$, as well as the tags that a user commonly assigns given by $\mathbf{A}_{TU}$. Tag recommenders that combine recommendations from both sources in one way or another have been developed for instance by [JMH+07, WSZ09, GBM+09]. Hybrid solutions are also required for other recommendation scenarios with more than one input object such as personalized social search [XBF+08]. We will present a novel recommendation method that can be applied to tag recommendation as well as personalized social search in Chapter 8.

Yet another hybrid approach is the FolkRank algorithm proposed by Hotho et al. [HJSS06a]. We discuss this algorithm seperately, since it has become the de facto benchmark for folksonomy reommendation of any type.

Adapted PageRank and FolkRank

Hotho et al. [HJSS06a] propose an extension of the well-known PageRank algorithm [BP98] to folksonomies. Instead of the web graph, they merge all node sets to a set $V = I \cup T \cup U$. They then construct a $|V| \times |V|$ weighted adjacency matrix $\mathbf{A}_{FR}$ where each entry corresponds to the number of times two nodes co-occur in Y. The matrix $\mathbf{A}_{FR}$ thus consists of all bipartite subgraphs of G:

$$\mathbf{A}_{FR} = \begin{pmatrix} 0 & \mathbf{A}_{IT} & \mathbf{A}_{IU} \\ \mathbf{A}_{TI} & 0 & \mathbf{A}_{TU} \\ \mathbf{A}_{UI} & \mathbf{A}_{UT} & 0 \end{pmatrix}.$$

The authors now assume a weight diffusion process that resembles the *random surfer* concept behind the PageRank algorithm [BP98]. This concept assumes a user that randomly follows the links of the web graph. Web resources are then weighted by the probability that they will be reached by this *random surfer*. However, instead of the web graph, Hotho et al. consider a user who browses along the undirected edges of the folksonomy hypergraph. A user is allowed to navigate from items to tags and users, from tags to users and items, and from users to items and tags, which simulates the discovery process commonly found in folksonomies and described in Section 1.3. The weight update for the **Adapted PageRank** process is defined as

$$\mathbf{w}_{t+1} \leftarrow d\mathbf{A}'_{FR}\mathbf{w}_t + (1-d)\mathbf{p},$$

where $\mathbf{A}'_{FR}$ is the row normalized, stochastic version of the adjacency matrix $\mathbf{A}_{FR}$, $\mathbf{p}$ is a preference vector that will generally represent context, and d regulates the decay of weight diffusion and thus the impact of the preference vector. The authors propose to set all entries of $\mathbf{p}$ to 1 to obtain a global ranking without context. To include context the authors suggest to set all entries of $\mathbf{p}$ to 1 except for the entries which correspond to the nodes of the context that are set to $|I|$, $|T|$, or $|U|$, respectively [JMH+07].

The fact that edges within folksonomies are undirected leads to relatively trivial results for the Adapted PageRank algorithm [HJSS06a, JMH+07], since weights flow back and forth between the same nodes, such that popular objects are likely to remain popular. This is not the case for the original PageRank scenario, since web links are directed. To overcome these intrinsic shortcomings of their algorithm, the authors propose a differential approach [HJSS06a]

$$\mathbf{w} = \mathbf{w}^1 - \mathbf{w}^0,$$

where $\mathbf{w}^0$ is the global weighting, that is obtained when $\mathbf{p} = \mathbf{1}$, and $\mathbf{w}^1$ is the result vector of the Adapted PageRank algorithm with $\mathbf{p}$ set to reflect the given context, as described above. The authors call this differential method **FolkRank**. It has been shown that the FolkRank algorithm performs consistently better than Adapted PageRank [HJSS06a, JMH+07, RBMNST09, WZB10b], since it better respects the actually given context.

The beauty of the FolkRank algorithm is its general applicability. The weight vector $\mathbf{w}$ contains weights for all objects independent of their type, and it is possible to construct a preference vector $\mathbf{p}$ for all kinds of contexts. This makes the FolkRank method applicable to all the recommendation scenarios that we identified in Section 6.2. However, this universality is generally traded for less accurate recommendations when compared to more specialized solutions [RBMNST09, SNM08].

Tensor approaches

All presented recommendation methods so far require the reduction of the folksonomy hypergraph G to bipartite subgraphs. On the contrary, tensor-based approaches work directly on the hypergraph G. This lets them avoid the loss of information caused by all forms of graph reduction. The authors of

Table 6.2: Existing approaches to folksonomy recommendation and the data structures they are built upon. The majority of approaches reduce the folksonomy hypergraph to a set of bipartite subgraphs, which looses information.

Approach	$\boldsymbol{\mathcal{Y}}_{ITU}$	$\mathbf{A}_{IU}$	$\mathbf{A}_{IT}$	$\mathbf{A}_{TU}$
Tag-based item models			X	
Collaborative Filtering				
– user-centric tag-based				X
– user-centric item-based		X		
– item-centric		X		
Hybrid				
– tags as intermediates			X	X
– simple tag recommendation			X	X
– tag-enhanced collaborative filtering				
+ item-centric, e.g. HyPLSA (Chapter 7)		X	X	
+ user-centric		X		X
– Adapted Pagerank/FolkRank		X	X	X
Tensor based				
– tensor decomposition	X			
– 3-mode probabilistic models, e.g. [WZY06]	X			
Translation model (Chapter 8)	X		X	

[SNM08] apply Higher Order Singular Value Decomposition (HOSVD) in order to approximate the original folksonomy tensor by a tensor or lower rank. Similar to 2-dimensional Singular Value Decomposition (SVD), HOSVD produces weight estimators for all tuples of the original tensor $\boldsymbol{\mathcal{Y}}_{ITU}$. This allows the prediction of the most likely edges for a given context. Rendle et al. [RBMNST09, RST09] achieve an even better performance by using ranking-based tensor factorization models. Their model does not try to approximate values for all missing entries in $\boldsymbol{\mathcal{Y}}_{ITU}$, but rather identifies valid tag rankings for each user-item combination. We will present an additional tensor-based approach to the tasks of tag recommendation and tag-based social search in Chapter 8.

Table 6.2 gives an overview of existing approaches to graph-based folksonomy recommendation and the graph structures these are built upon.

Beyond graph-based methods

In this thesis we solely consider recommenders that are based on the folksonomy graph G. However, there has been a variety of work on folksonomy recommendation beyond purely graph-based methods. The most common extension is the combination of item tags and words from the content of an item into a single model [CBC08, dGLSB08]. This unification was shown to improve the quality of tag recommendations especially in cold-start situations, where an item has not yet received many tags [HKGM08, RHMGM09]. Similarly, tag recommenders that consider item meta-data, such as the title of an URL or a publication, during training were shown to outperform purely graph-based methods [TSD08].

Additional information may also be offered by external sources. For instance, an enhanced quality of tag-based item models can be achieved by incorporating website classifications and descriptions from external services, such as the Open Directory Project [SVG08]. External information was also shown to help in the design of better tag models, e.g. by grouping tags using external sources, such as WordNet [OGC08]. Finally, in addition to external information about tags and items it is also possible to integrate external knowledge about users, e.g. about their activity in other services [SCA08].

The remainder of this work presents three novel approaches to different aspects of folksonomy recommendation. Chapter 7 introduces a hybrid probabilistic recommender that merges usage and tagging patterns into a unified model for high quality item recommendations. Chapter 8 then looks at tagging from a user-centric perspective and proposes a translational approach for the mapping of personomy tags to a global and consistent vocabulary. This translational method leads to a better understanding of the concepts that each user associates with the tags from her personomy, which in turn permits the design of highly personalized tag and item recommenders. Finally, we present a novel probabilistic method for the challenge of tag-based trend detection within folksonomies in Chapter 9. All proposed solutions perform favorable compared to the existing *state-of-the-art*.

Chapter 7

A hybrid approach to item recommendation

In this chapter, we consider the problem of personalized item recommendation in folksonomies. Our intent is to help folksonomy users discover interesting items based on their bookmarking history. To this end, we exploit the semantic contribution of tags and extend a classical collaborative filtering approach by user-generated annotations. Our hybrid probabilist recommender is built in the spirit of the Probabilistic Latent Semantic Analysis (PLSA) and merges usage and tag observations into a single model. Extensive experiments on three real-world folksonomy datasets show that the incorporation of tags drastically improves recommendation results. Furthermore, we find the Hybrid PLSA (HyPLSA) recommender to produce higher quality recommendations than existing *state-of-the-art* approaches, such as FolkRank or Singular Value Decomposition (SVD).

Probabilistic latent semantic analysis (PLSA), as introduced by Hofmann [Hof99b], was shown to improve recommendation quality in various settings by assuming a lower dimensional latent topic model as the origin of the observed co-occurrence distributions. Our HyPLSA recommender extends the classical PLSA model and unifies the observations contained in the bipartite co-occurrence matrices $\mathbf{A}_{IU}$ and $\mathbf{A}_{IT}$, thus combining collaborative and annotation-based evidences. We explicitly exclude the $\mathbf{A}_{TU}$ matrix, which contains the personomies of all users, since user-level differences in tagging strategies render personomy-based user comparisons problematic (see Chapter 8). Our approach is algorithmically related to previously presented work on hybrid recommender systems that combine collaborative and content-based models for better recommendation quality, see e.g. [CH00, PUPL01]. However, instead of relying on the often complex, hard to extract, and possibly heterogeneous content of items, we only consider an item's annotations.

Especially young communities that want to offer recommendation functionality to their users suffer from lack of historical data. This problem is generally referred to as the *cold-start* problem. At this early stage it is difficult to derive user or item similarities based on collaborative techniques only. However, as noted before, tags act as intermediates between other object types, e.g. connecting items and users, and can thus help build more robust recommendation models. This is especially appealing in cold-start related situations, since, as extensively discussed in Chapter 3, objects will more likely show overlapping patterns in their respective tag distributions than in the usage dimension. We therefore investigate the performance of our HyPLSA approach in such cold-start environments in a dedicated section. Our experiments show that recommendation quality can be drastically improved when considering tags.

This chapter is structured as follows: The first section describes the concepts behind the probabilistic latent semantic analysis, as introduced by Hofmann [Hof99b]. Section 7.2 then presents the HyPLSA recommender that extends the classical PLSA concept to folksonomies. This recommender is evaluated in two scenarios. The first scenario considers the task of personalized item recommendation in mature folksonomies and is discussed in Section 7.3. The second scenario instead elaborates the beneficence of our approach in cold-start environments and is presented in Section 7.4.

7.1 Probabilistic latent semantic analysis

The aspect model of PLSA associates the co-occurrence of observations with a hidden topic variable $Z = \{z_1, \ldots, z_k\}$. In the context of document retrieval, it assumes that documents deal with a mixture of unobserved topics which in turn are responsible for the generation of words. Similar to other dimensionality reduction methods, such as SVD, PLSA aims to map terms and documents to a lower dimensional latent space in order to reduce noise- and sparsity-related problems as well as to lower computational efforts. Documents and words are assumed to be statistically independent once the generating topic mixture is known. The probability that a word w and a document d co-occur can hence be calculated by summing over all latent topics:

$$P(d, w) = P(d)P(w|d) = P(d) \sum_{Z} P(w|z)P(z|d). \tag{7.1}$$

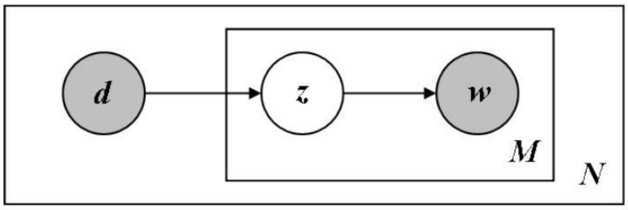


Figure 7.1: Plate notation of the document-word PLSA aspect model. Words are drawn from an unobserved set of latent topics Z which in turn are drawn for each document. Documents and words are assumed independent given Z.

The plate notation of this generative model is presented in Figure 7.1. The log-likelihood of the observed co-occurrence data can be calculated as

$$L = \sum_D \sum_W f(d, w) \log P(d, w), \tag{7.2}$$

where $f(d, w)$ denotes the number of times word w occurs in document d. Combining Equations 7.1 and 7.2 the log-likelihood of the data given the latent topic model can be calculated as

$$L = \sum_D \sum_W f(d, w) \log \sum_Z P(w|z)P(z|d)P(d). \tag{7.3}$$

As described in [Hof99b], this log-likelihood can be maximized by applying the Expectation-Maximization (EM) algorithm that produces estimators for $P(w|z)$ and $P(z|d)$[1]. The estimated probability $P(d, w)$ from Equation 7.1 can then be used to retrieve the documents which are most similar to a query $q = \{w_1, \ldots, w_N\}$ within the latent topic space. Furthermore, it is possible to estimate document and word similarities by comparing the latent topic distributions $P(z|w)$ and $P(z|d)$, where $P(z|w)$ can be calculated from $P(w|z)$ using Bayes rule.

One of the major challenges when deriving lower dimensional models is the estimation of the number of latent topics K. Choosing too few topics may not fully reflect the underlying latent composition of a domain. On the other hand, modeling too many topics may lead to an overfitting on the training data. We will therefore evaluate the performance of our approaches for different values of K.

[1] The document probability $P(d)$ from Equation 7.3 can be neglected during this maximization process, since it only adds a constant value to L.

7.2 HyPLSA: Hybrid PLSA for folksonomies

We build our hybrid model by reducing the folksonomy tensor $\boldsymbol{\mathcal{Y}}_{ITU}$ to two bipartite matrices $\mathbf{A}_{IU}$ and $\mathbf{A}_{IT}$. These matrices contain item-user and item-tag co-occurrence counts. We do not include the tag-user matrix $\mathbf{A}_{TU}$, since the heterogeneity in users' tagging behaviors impedes direct personomy comparisons (see Chapter 8). From a collaborative filtering perspective, an observation corresponds to the bookmarking of an item by a user. All collaborative filtering observations are hence given by the elements of the matrix $\mathbf{A}_{IU}$. In the spirit of PLSA, we assume that item and user events occur independently given the topic variable Z. Similar to document retrieval, the conditional probability that a given user bookmarks an item can be computed by summing over all latent variables:

$$P(i|u) = \sum_{Z} P(i|z)P(z|u). \tag{7.4}$$

We now assume the same hidden topics as origins of the item-tag co-occurrence data given by the matrix $\mathbf{A}_{IT}$. Analogous to Equation 7.4, the conditional probability between tags and items is given as:

$$P(i|t) = \sum_{Z} P(i|z)P(z|t). \tag{7.5}$$

Following the procedure in [CH00], we can now combine both models based on the common factor $P(i|z)$ by maximizing the log-likelihood function

$$\begin{aligned} L \;=\; & \sum_{I} \Bigg[\alpha \sum_{U} \mathbf{A}_{iu} \log \sum_{Z} P(i|z)P(z|u) \\ & + (1-\alpha) \sum_{T} \mathbf{A}_{it} \log \sum_{Z} P(i|z)P(z|t) \Bigg]. \end{aligned} \tag{7.6}$$

The α parameter ($0 \leq \alpha \leq 1$) is a predefined weight for the influence of each two-mode model. The $\mathbf{A}_{iu}$ and $\mathbf{A}_{it}$ terms denote the co-occurrence counts of an item with a user or tag, and thus correspond to the $f(d, w)$ term of Equation 7.2. The plate notation of this hybrid model is shown in Figure 7.2. Using the Expectation-Maximization (EM) algorithm [CH00], we then perform maximum likelihood parameter estimation for the aspect model. During the expectation (E) step we first calculate the posterior probabilities:

$$P(z|u,i) = \frac{P(i|z)P(z|u)}{P(i|u)} \tag{7.7}$$

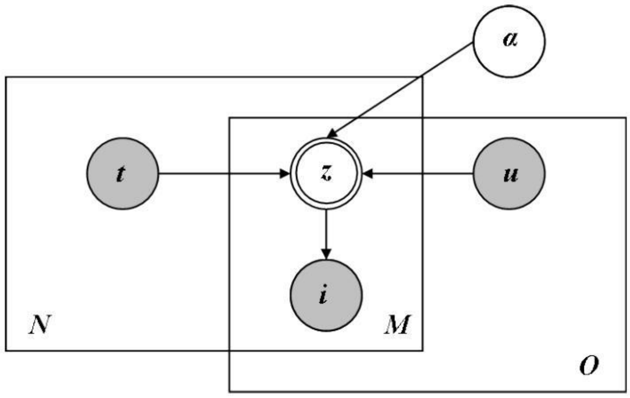


Figure 7.2: Plate notation of the Hybrid PLSA aspect model. The hybrid model is built by combining two basic PLSA models, assuming the same latent topics as origins for item-user as well as item-tag co-occurrences. The mixture ratio is controlled by the α parameter.

$$P(z|t,i) = \frac{P(i|z)P(z|t)}{P(i|t)}, \tag{7.8}$$

and then re-estimate parameters in the maximization (M) step as follows:

$$P(z|u) \propto \sum_I \mathbf{A}_{iu} P(z|u,i) \tag{7.9}$$

$$P(z|t) \propto \sum_I \mathbf{A}_{it} P(z|t,i) \tag{7.10}$$

$$\begin{aligned} P(i|z) \propto\ & \alpha \sum_U \mathbf{A}_{iu} P(z|u,i) \\ & + (1-\alpha) \sum_T \mathbf{A}_{it} P(z|t,i). \end{aligned} \tag{7.11}$$

Based on the iterative computation of the above E and M steps, the EM algorithm monotonically increases the likelihood of the combined model on the observed data. The α parameter controls the influence of users versus tags. Setting $\alpha = 1$ reduces the model to a purely collaborative filtering recommender, whereas $\alpha = 0$ leads to a purely tag-based model. Items are now recommended to a user u based on the probability $P(i|u)$ from equation 7.5[2]. We set $P(i|u)$ to 0 for items a user already bookmarked, since users can bookmark items only once.

[2]It is likewise possible to recommend items with respect to a given tag t based on Equation 7.5.

There exist a variety of methods that try to avoid the convergence of the PLSA model to suboptimal local maximums. The most commonly used method is probably Tempered Expectation-Maximization (TEM), as proposed in the original work by Hofmann [Hof99b]. However, due to better performance in our own evaluations, we do not apply TEM, but follow an optimization approach taken by Brants et al. [BCT02]. The authors note that, since the PLSA model converges to a local maximum, different random initializations of the EM algorithm will result in different locally optimal models. They further report that averaging the estimates from models that evolved from different initializations helps to overcome local locked-in effects and produces more robust weights. In all our evaluations we therefore train each HyPLSA model 10 times on the same data, but with random EM initializations. We then average the probabilities $P(i|u)$ over all 10 models.

The probabilistic model of the HyPLSA recommender maps items, tags, and users to the same latent topic space Z. This consistent representation enables us to calculate similarities between all types of objects. This universality of our approach allows its application to all recommendation scenarios, including (personalized) item recommendation, tag-aware social search, expert finding, and even tag recommendation 6.1. In this thesis, we will mainly discuss its benefits for the challenge of personalized item recommendation, and will evaluate the performance of our HyPLSA recommender in two relevant scenarios. The first scenario is classical item recommendation without any temporal considerations. The second scenario instead simulates the cold-start phase of a folksonomy and investigates how recommendation quality changes once the community starts to grow.

7.3 Scenario 1: Personalized item recommendation

As introduced earlier, the task of personalized item recommendation is to predict items from a known set I that a given user may be interested in. We thus want to predict $\mathbf{A}_{Iu}$ for a given user u based on incomplete training data. Solving this task requires a profound understanding and correct modeling of a user's general preferences. Using our HyPLSA model, we describe each user's interests as a mixture of latent topics and recommend the items that are most closely related to these topics. This is equivalent to the ranking of items based on the probability $P(i|u)$.

Experiments

We conduct our experiments on snapshots of two real-world folksonomies. The *bibsonomy*$_{2009}$ dataset has been described in Section 1.5. From the spam filtered *delicious*$_{clean}$ dataset we only consider bookmarking events that fall before January 1, 2005. We follow common practice and perform our evaluation on p-core versions of the original graphs, with p set to 5. The resulting global dataset statistics are shown in Table 7.1. Despite the p-core reduction, we find the corresponding co-occurrence matrices $\mathbf{A}_{IT}$ and $\mathbf{A}_{IU}$ to remain rather sparse. Only 0.010 ($\mathbf{A}_{IU}$) and 0.022 ($\mathbf{A}_{IT}$) percent of the Bibsonomy's matrices contain non-zero entries, whereas for the Delicious data these percentages amount to 0.001 and 0.001, respectively.

Table 7.1: Statistics for both datasets and p-cores.

	p	TAS	Posts	Items	Tags	Users
Bibsonomy	1	1,401,104	421,928	378,378	93,756	3,617
	5	55,008	13,973	1,755	1,584	358
Delicious	1	3,906,207	1,909,687	982,356	140,645	25,966
	5	1,404,101	656,159	42,058	13,204	13,757

For each p-cored dataset we randomly select 90 percent of all bookmarks for training and keep the remaining 10 percent for testing. The bookmarks from the training data are then used to construct the co-occurrence matrices $\mathbf{A}_{IT}$ and $\mathbf{A}_{IU}$. We then train 10 HyPLSA models for each setting of α and K, and average the estimates of $P(i|u)$ over all models in order to avoid local maxima. For each bookmark $B_{i'u'}$ of the test set we then recommend items sorted by $P(i|u')$. We set $P(i|u') = 0$ for items that a user has already bookmarked during the training phase. The quality of our suggestions is evaluated by looking at the positions of the test items within the recommended item list. Lower rankings indicate a better recommendation quality. We determine this quality by different standard evaluation measures, such as Mean Average Precision (MAP) or rank-dependent Precision and Recall measures. All measures are average over all test bookmarks. Furthermore, in order to get more significant results, we repeat the evaluations 5 times and average the results over all runs. All obtained results are compared to a selection of baseline recommenders:

MostPopular This recommender weights items by their global popularity without any personalization. Most-popular recommenders have become a standard feature of most Web 2.0 services.

Adapted PageRank and FolkRank Both recommenders are designed as described in Section 6.2, with d set to 0.7 and the number of iterations set to 10. This resembles the setting in [HJSS06a]. Other parameter settings do not yield significant performance improvements.

Collaborative Filtering SVD The Singular Value Decomposition (SVD) has become a standard technique in many recommendation and search tasks [SKKR02, SKKR00, DDF+90, Bra03]. It allows to approximate a matrix $\mathbf{M}$ by a lower-rank matrix $\tilde{\mathbf{M}}$. This is achieved by factorizing the original matrix as product $\mathbf{M} = \mathbf{U\Sigma V}'$, where $\mathbf{\Sigma}$ is a diagonal matrix that contains the singular values of $\mathbf{M}$, and the matrices $\mathbf{U}$ and $\mathbf{V}'$ are input-output transformation matrices [GR70]. $\mathbf{M}$ can be approximated by the matrix $\tilde{\mathbf{M}}$ of rank K that results from $\tilde{\mathbf{M}} = \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}'$, where $\tilde{\mathbf{\Sigma}}$ is the same as $\mathbf{\Sigma}$ except that values that do not fall among the K highest singular values of $\mathbf{\Sigma}$ are set to 0. This minimizes the Frobenius norm of the difference between both matrices $||M - \tilde{M}||_F$ for a given value of K[3]. SVD is especially appealing for recommendation scenarios, since it produces estimates for unobserved co-occurrence events represented by zero values in the original matrix $\mathbf{M}$. For the same reasons, SVD has become a popular method for document modeling, where the decomposition is generally performed on a document-term co-occurrence matrix [DDF+90].

In a collaborative filtering scenario, we apply SVD to approximate the matrix $\mathbf{A}_{IU}$ by a matrix $\tilde{\mathbf{A}}_{IU}$ of rank K. For a given user u we can then recommend items based on their weight in the vector $\tilde{\mathbf{A}}_{Iu}$. These weights can be calculated within the factorized space without the computationally expensive construction of the dense matrix $\tilde{\mathbf{A}}_{IU}$.

Tag-based SVD The tag-based SVD recommender performs singular value decomposition to estimate the matrix $\mathbf{A}_{IT}$ by the matrix $\tilde{\mathbf{A}}_{IT}$ of rank K. If we choose the factorization of $\mathbf{A}_{IT}$ where the diagonal entries of $\mathbf{\Sigma}$ are sorted in decreasing order and remove the columns and rows of $\mathbf{U}$ and $\mathbf{V}$ that correspond to zero entries within $\tilde{\mathbf{\Sigma}}$, we can now write $\tilde{\mathbf{A}}_{IT} = \mathbf{U}_{IK}\tilde{\mathbf{\Sigma}}_{KK}\mathbf{V'}_{TK}$. This allows us to calculate the similarity between two items by comparing their eigenvalue spectra in the reduced space which can be achieved by simple matrix multiplication [DDF+90]

$$\mathbf{Q}_{II} = \mathbf{U}_{IK}\tilde{\mathbf{\Sigma}}^2_{KK}\mathbf{U}'_{IK}.$$

[3]The Frobenius norm of a matrix $\mathbf{A}_{XY}$ is defined as $||\mathbf{A}_{XY}||_F = \sqrt{\sum_X \sum_Y \mathbf{A}^2_{xy}}$.

Each entry of $\mathbf{Q}_{II}$ thus corresponds to the similarity between two items defined as the dot product between their respective vectors in the reduced space. For each user we then recommend the items that are most similar to her previously bookmarked items by calculating

$$\mathbf{w}(u) = \mathbf{A}'_{Iu}\mathbf{Q}_{II}.$$

Since $\mathbf{A}_{IU}$ is a boolean matrix ($\mathbf{A}_{iu} \in \{0,1\}$), each item weight is given as $w_i(u) = \sum_{j \in I_u} \mathbf{Q}_{ij}$.

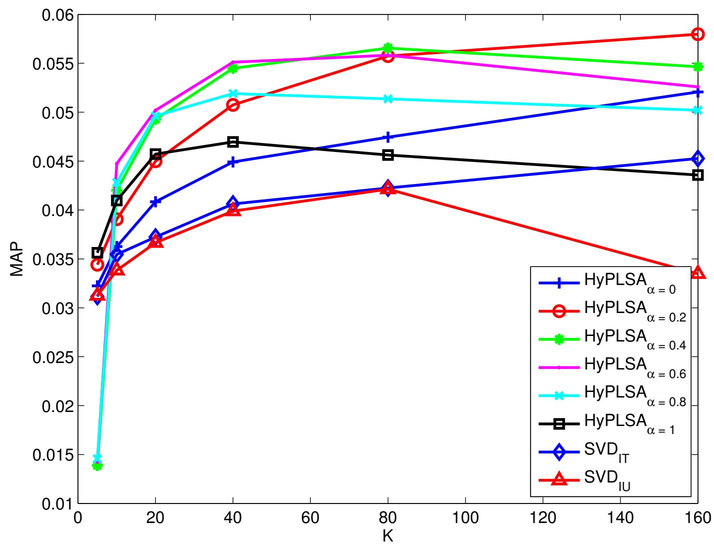
As for the PLSA recommender, we set the weights of previously bookmarked items to 0 and measure the position of bookmarked test items in the recommendation list.

Results

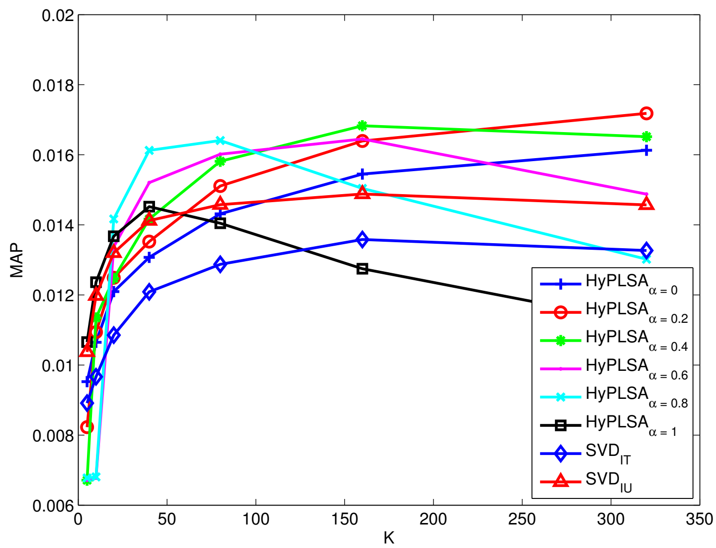
Optimal parameter settings

Figure 7.3 compares the mean average precision (MAP) values achieved by different HyPLSA recommenders with varying values of α and K. The plots additionally include the results for collaborative filtering and tag-based SVD baseline classifiers (SVD_{IU} and SVD_{IT}). We observe very similar tendencies on both datasets. The purely collaborative HyPLSA model ($\alpha = 1$) performs rather poorly and starts to overfit for values of $K > 40$. On the other hand, we find that the purely tag-based HyPLSA recommender ($\alpha = 0$) does not suffer from overfitting, but constantly improves with increasing values of K. These characteristics on both extremes of our model seem to become mingled in hybrid settings with $0 < \alpha < 1$. In these cases, we find models with higher collaborative contribution to overfit earlier compared to models that favor tags. The best performance is achieved for $K = 160$ (Bibsonomy) and $K = 320$ (Delicious). For models of these sizes we find a mixture ratio of $\alpha = 0.2$ to perform best. However, lower values of K may require a different mixture weight. The improved accuracy of the mixture models compared to purely tag-based or collaborative solutions demonstrates the validity of our hybrid approach. The same observation can be made when comparing the recommendation quality of our HyPLSA model to SVD-based approaches. Here, the best HyPLSA model outperforms both SVD recommenders by at least 30 percent in terms of MAP. This quality gain is independent of the number of latent topics.

One striking observation is the reproducibility of the described results on both datasets. The only major difference is a shift of patterns towards a higher number of latent topics K for the larger Delicious corpus. Alone the



(a) Bibsonomy



(b) Delicious

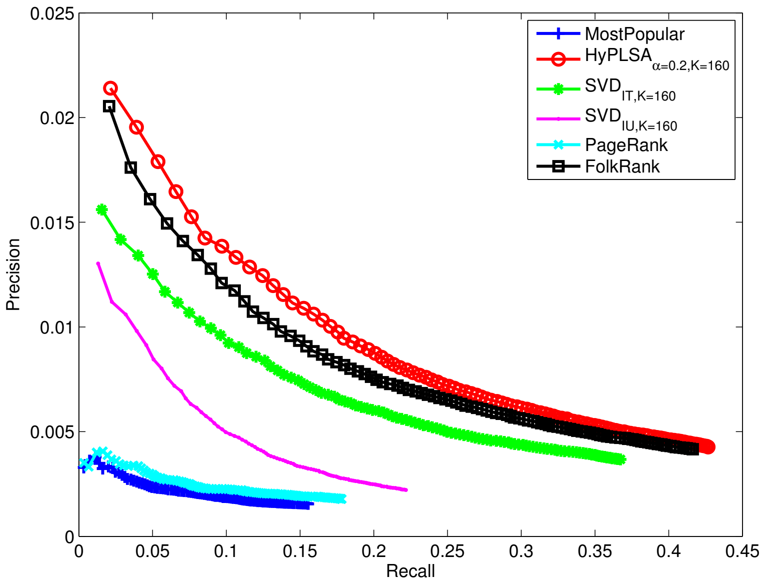
Figure 7.3: Mean average precision (MAP) as function of latent classes for various Hybrid PLSA and SVD recommenders. PLSA recommenders based on collaborative filtering only ($\alpha = 1$) tend to overfit already for small K. The purely tag-based PLSA recommender ($\alpha = 0$) requires more latent classes for the same recommendation quality, but does not suffer from overfitting. The best performance on both datasets is achieved by the HyPLSA recommender with a high number of latent classes K and $\alpha = 0.2$.

SVD recommenders seem to behave differently on both datasets. Whereas tag-based SVD (SVD_{IT}) performs favorably on Bibsonomy, we find the collaborative filtering SVD (SVD_{IU}) to be ahead on the Delicious dataset. One explanation for this difference is the relatively low overlap between items in the Bibsonomy dataset, where the most popular item occurs only 53 times in the p-core and 66 times in the full dataset. Users of Bibsonomy thus have a low likelihood to overlap in the item space, leading to poor collaborative filtering results.

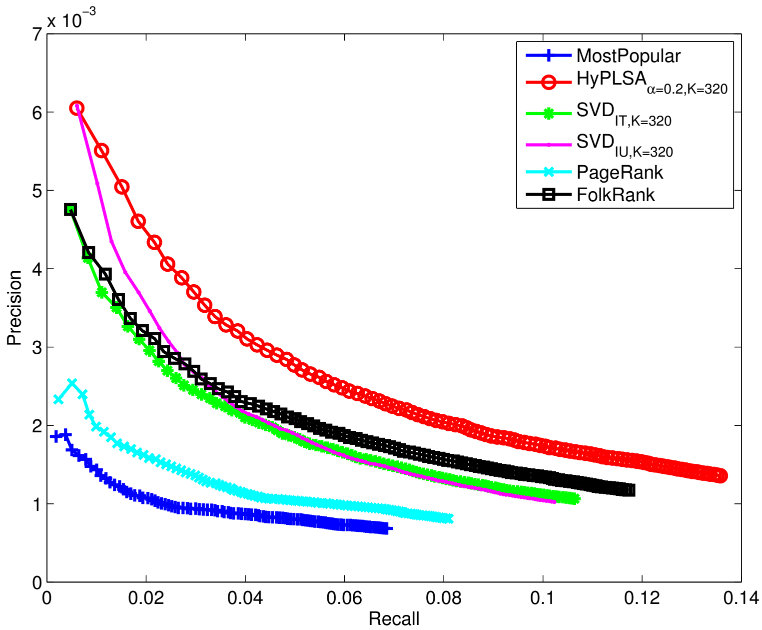
Comparison to state-of-the-art methods

Figure 7.4 displays the precision/recall plots for the best HyPLSA and SVD recommenders as well as for a variety of other recommender types. Once again, the plots show the dominance of the HyPLSA method independent of the dataset. Only for the Delicious dataset we find the collaborative SVD recommender ($SVD_{IU,K=320}$) to perform equally well on the first rank, but considerably worse on the consecutive ranks.

Different result patterns appear for the FolkRank algorithm. The method performs reasonably well on the Bibsonomy dataset, by far outperforming both SVD-based methods. However, this result is not consistent over both datasets. Instead, for the larger Delicious corpus we find the FolkRank algorithm to perform at the same level as SVD and drastically worse than the Hybrid PLSA method. The reason for this behavior is not clear. One possible explanation are scalability problems that lead to less accurate results on larger datasets. Since the Adapted PageRank method performs slightly better for Delicious, these problems could originate from the differential approach of FolkRank. The weighting of objects based on the difference between their relevance to a user and their global relevance tends to move very popular objects with limited relevance to a user to the end of the recommendation list. However, this may not fully reflect a user's preferences, since a very popular item, even if not considered very relevant for a user, is still more likely to be found within a user's item collection than a very unpopular one. Consequently, we find that FolkRank tends to return relevant items at the beginning and at the end of the recommended list, whereas irrelevant items end up in the middle of the result list. This effect is demonstrated very well by looking at the ROC curve of the FolkRank algorithm on the Delicious dataset (Figure 7.5). The curve shows a standard, i.e. above random, behavior for the first recommendations and an inverted, i.e. below random, behavior at the end of the recommendation list. This supports our assumption that many relevant items are ranked at the end of the recommended list due to the differential nature of the algorithm. This effect is not limited to



(a) Bibsonomy



(b) Delicious

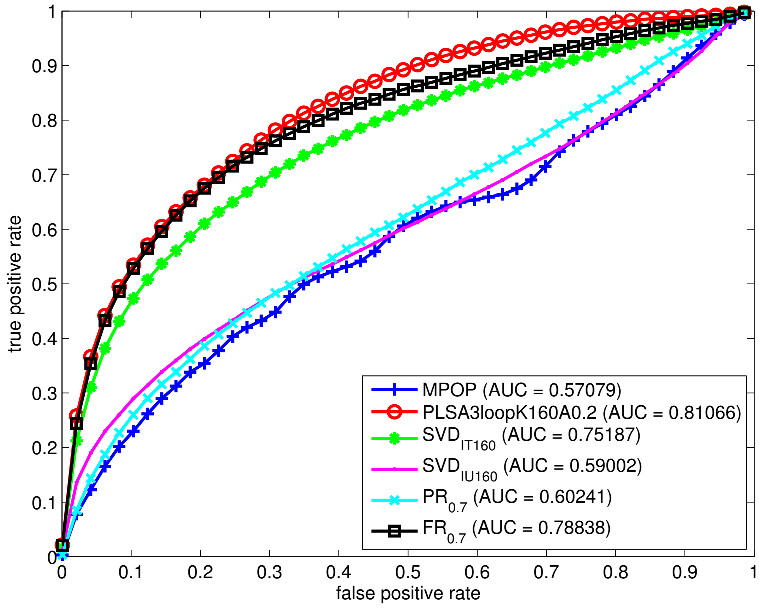
Figure 7.4: Precision/Recall curves for various recommenders on the task of Top 100 item recommendation. The Hybrid PLSA approach produces the best recommendation quality, outperforming other *state-of-the-art* techniques, such as FolkRank or SVD.

the FolkRank algorithm alone, but can also be observed for both SVD-based recommenders on the larger Delicious dataset.

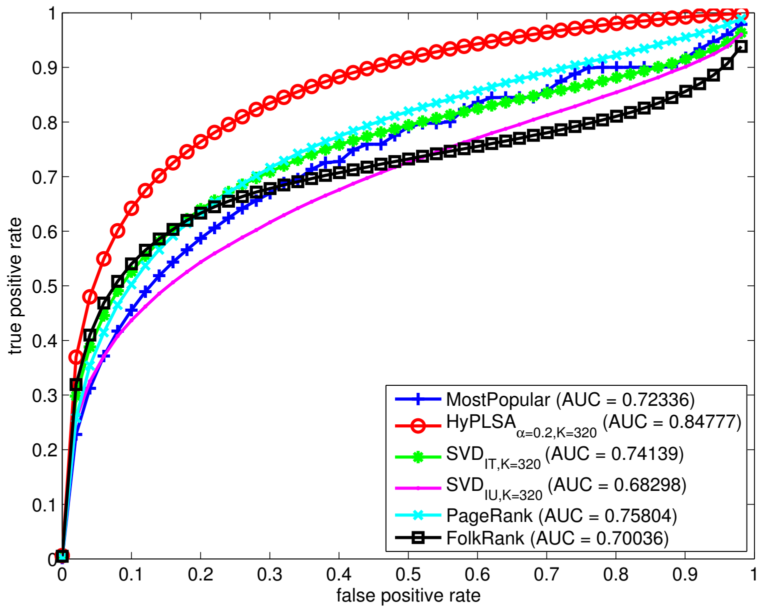
Except for the Adapted PageRank method, we find all recommenders to achieve considerably better results than the MostPopular baseline method on the first 100 ranks (Figure 7.4). On the contrary, when evaluating the quality of the entire recommendation list, we find that some approaches do not even reach the quality achieved by this baseline method in terms of Area under Curve (AUC), as illustrated in Figure 7.5. However, the quality of the recommendation tail will be of little relevance in the majority of scenarios, since the amount of recommendations that can be presented to a user is limited. The Hybrid PLSA recommender performs better than all competitors in terms of Precision and Recall, MAP as well as AUC. This dominance occurs independently of the dataset, but is more pronounced on the larger Delicious corpus.

What do topics look like?

We conclude our result discussion by investigating the actual clusters generated by our HyPLSA model. Table 7.3 presents the most prominent tags and URLs for the first ten topics of a HyPLSA recommender that has been trained on Delicious with $\alpha = 0.2$ and $K = 80$. The topics of both tables refer to the same latent classes. We find that the clustering of tags works rather well and most latent topics seem to represent real world concepts, such as "programming" (Topic 3), "shopping" (7), "web development" (9), or "news" (10). However, other topics seem to mix real world concepts, e.g. "flash" and "semantic web" (1) or "perl" and "web services" (6). This indicates an insufficient number of latent topics K. The same patterns can be observed for the most popular URLs per topic. The overlapping concepts will generally lead to suboptimal recommendation performance. For instance, the PLSA model partly displayed in Table 7.3 will tend to recommend "web services"-related content to a user who is interested in "perl" or mix "flash"- and "semantic web"-related content whenever a user is mapped to Topic 1. One obvious way to overcome these shortcomings is to increase the number of latent topics K. This will result in models that better fit the trainings data, but risks overfitting as discussed previously. Further improvements can be achieved by averaging the obtained recommendation probabilities $P(i|u)$ over a set of independently trained models. This reduces the effect of overlapping concepts, since independently trained models are unlikely to share the same concept mixtures. As a result, we find such averaged models to produce higher quality recommendations.



(a) Bibsonomy



(b) Delicious

Figure 7.5: ROC curves for various recommenders on the task of item recommendation. The plots once more emphasize the quality improvements of a HyPLSA recommender compared to other *state-of-the-art* recommenders.

Topic	Highest ranked tags
1	flash, rdf, animation, semweb, foaf, calendar, semantic, semanticweb, actionscript, ical
2	bittorrent, p2p, torrent, torrents, latex, anime, download, bt, filesharing, tex
3	programming, .net, microsoft, c++, c, dotnet, development, c#, performance, mono
4	social, community, socialsoftware, collaboration, networks, network, research, web, social_software, sharing
5	security, hacking, hack, pvr, wifi, tivo, hacks, firewall, wireless, faq
6	perl, rest, webservices, api, programming, soap, amazon, cgi, chicago, services
7	shopping, shop, pdf, home, clothing, fashion, electronics, store, furniture, wishlist
8	design, inspiration, webdesign, portfolio, graphic, portal, graphics, illustration, nice, graphicdesign
9	html, web, standards, xhtml, accessibility, webdev, w3c, webdesign, presentation, webstandards
10	news, daily, magazine, imported, magazines, media, newspaper, newspapers, noticias, world

Topic 1	**Topic 6**
http://www.ii.uib.no/ arntzen/kalender/	http://www.netalive.org/tinkering/serious-perl/
http://www.amk.ca/talks/2003-03/	http://www.xml.com/pub/a/2004/12/01/restful-web.ht...
http://www.betaversion.org/ stefano/linotype/news/...	http://www.xfront.com/REST-Web-Services.html
http://www.maani.us/charts/index.php	http://naeblis.cx/rtomayko/2004/12/12/rest-to-my-w...
http://www.ferryhalim.com/orisinal/	http://www.xml.com/pub/a/2004/08/11/rest.html
http://www.foaf-project.org/	http://www.prescod.net/rest/mistakes/
http://hook.org/anselm/essays/20040915.htm	http://learn.perl.org/library/beginning_perl/
http://www.airtightinteractive.com/simpleviewer/	http://www.ccl4.org/ nick/P/Fast_Enough/
http://www.openlaszlo.org/	http://search.cpan.org/
http://www.laszlosystems.com/	http://www.perl.com/
Topic 2	**Topic 7**
http://www.suprnova.org/	http://www.thinkgeek.com/
http://www.slyck.com/news.php?story=596	http://www.naughtycodes.com/
http://www.bi-torrent.com/	http://www.howtofoldashirt.net/
http://pealco.net/archives/2004/11/08/how_to_never...	http://www.incompetech.com/beta/plainGraphPaper/
http://www.bitoogle.com/	http://www.threadless.com/
http://www.torrentreactor.net/index.php	http://www.ebay.com/
http://members.chello.nl/ p.wiersema/list.html	http://www.amazon.com/
http://www.blogtorrent.com/	http://www.newegg.com/
http://www.legaltorrents.com/	http://www.emachineshop.com/
http://www.donvitorrent.com/	http://www.pricewatch.com/
Topic 3	**Topic 8**
http://www.codeproject.com/	http://wellstyled.com/tools/colorscheme2/index-en....
http://www.cppreference.com/	http://www.k10k.net/
http://www.mono-project.com/about/index.html	http://www.intersmash.com/300images/
http://www.boost.org/	http://www.stylegala.com/
http://msdn.microsoft.com/	http://mocoloco.com/
http://www.gotdotnet.com/	http://www.ficml.org/jemimap/style/color/wheel.htm...
http://www.sgi.com/tech/stl/	http://www.cssbeauty.com/
http://csharp-source.net/	http://www.pixy.cz/apps/barvy/index-en.html
http://www.4guysfromrolla.com/	http://www.thinkingwithtype.com/
http://www.icsharpcode.net/OpenSource/SD/	http://37signals.com/papers/introtopatterns/
Topic 4	**Topic 9**
http://www.flickr.com/	http://www.meyerweb.com/eric/tools/s5/
http://www.orkut.com/	http://validator.w3.org/
http://www.lifewithalacrity.com/2004/10/tracing_th...	http://www.w3.org/
http://gmpg.org/xfn/	http://www.maxdesign.com.au/presentation/checklist...
http://shirky.com/writings/group_user.html	http://www.sitepoint.com/print/1273
http://www.audioscrobbler.com/	http://www.456bereastreet.com/archive/200410/bring...
http://www.corante.com/many/	http://daringfireball.net/projects/markdown/
http://www.foaf-project.org/	http://www.fckeditor.net/
http://socialsoftware.weblogsinc.com/	http://www.kryogenix.org/code/browser/sorttable/
http://www.infoworld.com/article/04/08/20/34OPstra...	http://www.usability.com.au/resources/forms.cfm
Topic 5	**Topic 10**
http://www.phenoelit.de/dpl/dpl.html	http://slashdot.org/
http://www.hackaday.com/	http://news.bbc.co.uk/
http://www.insecure.org/tools.html	http://www.cnn.com/
http://www.buzzsurf.com/surfatwork/	http://news.google.com/
http://johnny.ihackstuff.com/index.php?module=prod...	http://www.nytimes.com/
http://www.mythtv.org/	http://www.wired.com/
http://www.securityfocus.com/	http://www.fark.com/
http://www.byopvr.com/	http://www.guardian.co.uk/
http://www.snopes.com/	http://www.theregister.co.uk/
http://www.e-fense.com/helix/	http://www.drudgereport.com/

Table 7.2: Tags and URLs of highest co-occurrence probability with the first 10 latent topics of a HyPLSA recommender with $\alpha = 0.2$ and $K = 80$. Most topics seem to represent real-world concepts, e.g. "programming" (3), "social networking" (4), or "news" (10). Other latent topics seem to mix concepts, e.g. "flash" and "semantic web" (1) or "perl" and "web services" (6), indicating an insufficient number of latent topics K.

7.4 Scenario 2: Recommendation in cold-start environments

In a second scenario, we consider the problem of personalized item recommendation in young folksonomies, where little is still known about items, users, and tags alike. During this *cold-start* phase recommendations have to be made based on very little historical data. As a consequence, similarities between items or users based on usage information only are hard to calculate, since the $\mathbf{A}_{IU}$ matrix will suffer severely from sparsity. The incorporation of tags in this critical phase therefore seems especially appealing, since tags possess denser connectivity patterns, as reported in Chapter 2. Hence, our objective is to increase the quality of item recommendations during the cold-start phase of a folksonomy by utilizing item-tag co-occurrences and user-item co-occurrences in parallel. We realize this objective by applying our HyPLSA recommender.

Experiments

We conduct our experiments on the *citeulike* dataset described in Section 1.5. The first bookmark was posted in CiteULike on November 4, 2004. From the full CiteULike corpus we only consider the first 24 months in our evaluations, since we are interested in the cold-start phase of the service. The are many reasons for choosing CiteULike. They include the completeness of the dataset, its extensive use in previous research on folksonomies [OGC08, ZC08], as well as the existence of previous work on cold-start related questions [BvdB08] that was performed on similar data.

As in [BvdB08], we prune the original dataset by removing all users who have bookmarked less than 20 items as well as items and tags that occur only once. We then create monthly snapshots, where each of the snapshots accumulates all previous bookmarking events. In this way, we are able to simulate a growing folksonomy over time. Figure 7.6 presents monthly graph statistics of the resulting dataset.

For each month of the dataset, we randomly select 80 percent of each user's bookmarks for training and keep the rest for testing. We then train all recommenders on the training sets and evaluate their performance on the test sets. The HyPLSA recommenders are designed exactly as in the previous scenario. The relatively small size of the dataset allows us to optimize parameters per recommender and month through a brute-force approach. Results are averaged over all users and 10 independent test runs. We compare the results of our Hybrid PLSA recommender to the performance of

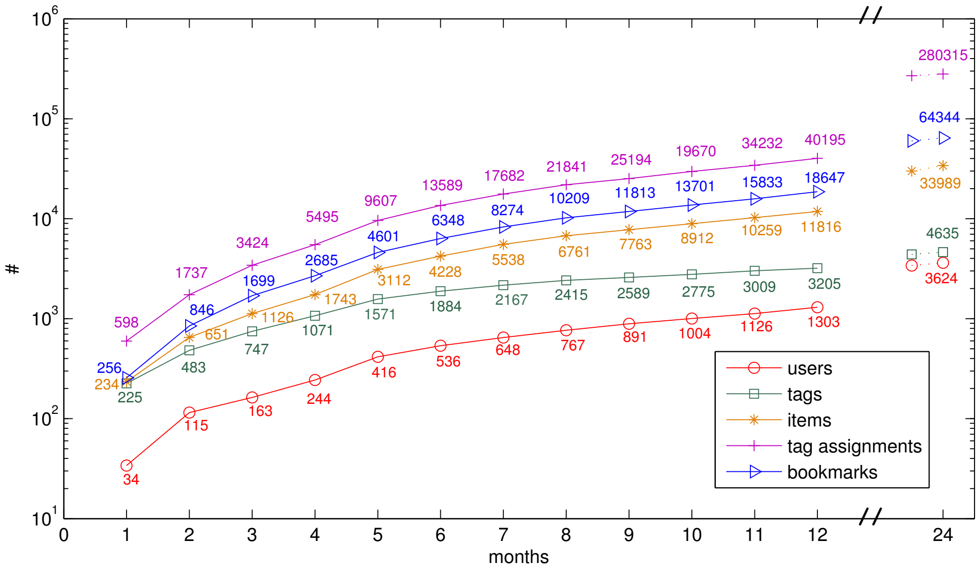


Figure 7.6: Statistics for the initial 24 months of the growing CiteULike dataset (log-scale) [SWUH09]. Each point corresponds to the size of a set after the respective number of months.

a MostPopular baseline as well as to the results achieved by the FolkRank algorithm. The latter is configured as described in the previous section.

Results

We evaluate the performance of each recommender based on the precision at 10 measure (Prec@10). Other evaluation measures, such as mean average precision (MAP) or F_1 scores, show similar results. Figure 7.7 displays the Prec@10 values achieved by the HyPLSA recommender with optimal parameter settings ($\alpha = 0.1, K = 80$), a purely tag-based PLSA recommender ($\alpha = 0, K = 80$), a purely collaborative PLSA model ($\alpha = 1, K = 10$) as well as the precision of the FolkRank and MostPopular methods. The figure illustrates that item usage information alone does not suffice to build qualitative models. In fact, we observe that a purely collaborative PLSA recommender already overfits for more then 10 latent classes during the initial months. For later months with more available data this overfitting occurs at $K = 20$ (not shown in figure). On the other hand, we find the purely tag-based model to perform rather well and to be significantly more accurate than FolkRank and purely collaborative methods. This further emphasizes the potential of tags for recommender systems. Improvements reach a peak during the fourth

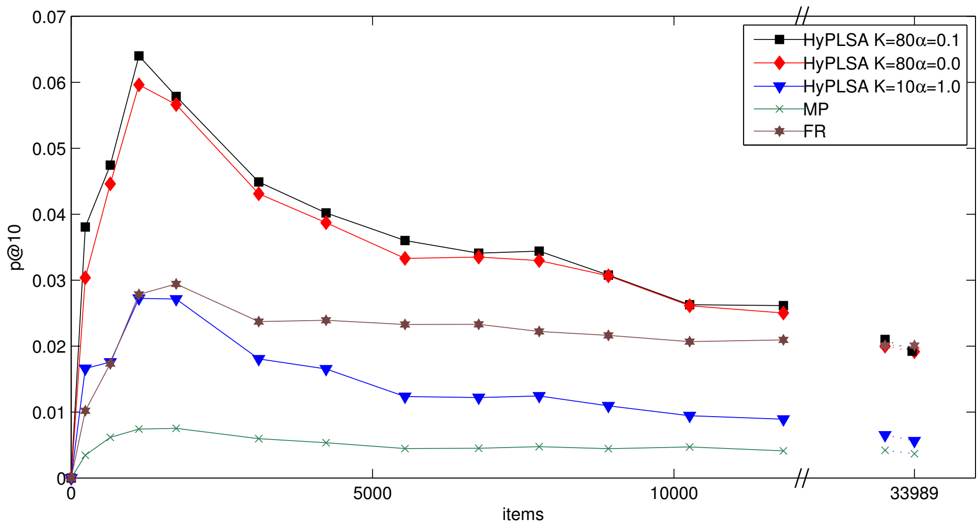


Figure 7.7: Precision@10 for different item recommenders during a simulation of the initial months of the *citeulike* service. Points correspond to monthly measurements [SWUH09].

and fifth months of the dataset when the annotation-based recommender returns twice as many relevant results as FolkRank. However, results can still be improved by up to 25 percent (month 2) when fusing collaborative and tag-based observations into a unified HyPLSA model with a strong tendency towards annotation-based observations ($\alpha = 0.1$). We also observe that all PLSA recommenders as well as the FolkRank method perform significantly above the MostPopular baseline.

When looking at temporal aspects of the result curves, we find recommendation quality to increase independent of the recommender once the folksonomy starts to grow. After the initial months, this improvement seems to be compensated by an increased complexity of the reommendation task, such that the constant increase in recommendable items leads to a drop in precision. We also find that the initial major quality gains of the annotation-based and hybrid methods are less significant once the folksonomy has reached a certain size. One possible explanation are increasing density patterns in the $\mathbf{A}_{IU}$ matrix that reduce the positive impact of tag incorporation. The same effect could also be caused by changing test set characteristics during later months, e.g. the appearance of very few highly active users that dominate the evaluation.

7.5 Chapter conclusions

In this chapter, we investigated how tags can be incorporated into existing recommendation approaches in order to improve recommendation performance. We presented a novel hybrid probabilistic approach to item recommendation in a folksonomy environment. The proposed method extends the well-known PLSA algorithm in order to fuse usage and tagging information into a unified model, leading to an improved descriptiveness. Moreover, our approach enables us to map items, tags, and users into the same latent topic space. This consistent modeling makes our HyPLSA approach universally deployable in all types of recommendation settings.

We evaluated the applicability of our approach to the task of personalized item recommendation in two potential application areas. The first scenario considered mature folksonomies, where a large amount of historical data exists. In the second scenario we instead examined a young folksonomy with a low amount of existing information and investigated the performance of our hybrid recommender during different stages of the cold-start phase. We performed extensive evaluations for both scenarios on the data of three different real-world folksonomies. The results showed that the proposed hybrid approach produces higher quality recommendations than standalone collaborative filtering or annotation-based methods. Furthermore, we find the HyPLSA recommender to compare favorably with existing *state-of-the-art* recommendation solutions, such as FolkRank or SVD.

One recurrent observation in all our experiments is a general quality gain when using tags compared to basic collaborative filtering models. These quality improvements become especially apparent in scenarios with limited information about item usage, as is the case during the cold-start phase of folksonomies and still seems to be true for the relatively small $bibsonomy_{2009}$ dataset. Furthermore, these improvements seem to occur independent of the method, such that we find similar patterns for PLSA and SVD derived models. These results highlight the potential of tags for the design of effective item recommenders. Collaborative methods only performed better than tag-based techniques for the larger Delicious dataset where the amount of historical data leads to more representative user models.

A major advantage of SVD-based methods is the availability of highly optimized software libraries. Furthermore, the fact that our PLSA approach needs to train many models in parallel in order to stabilize results linearly increases training time requirements. These factors slow down the training time of our HyPLSA implementation by a factor of 20 to 50 as compared to SVD models with the same number of latent topics. However, these costs in computation only occur during the off-line training, whereas runtime efforts remain mostly unaffected. The applicability of PLSA in large-scale recommendation scenarios was successfully demonstrated in [DDGR07].

In this chapter, we have presented a hybrid recommender in order to cope with the complex structure of folksonomies. Our method successfully reduced information loss by incorporating usage and tag-based observations into a unified model. However, the HyPLSA approach still requires the reduction of the folksonomy tensor $\boldsymbol{\mathcal{Y}}_{ITU}$ to two bipartite matrices $\mathbf{A}_{IU}$ and $\mathbf{A}_{IT}$. After this reduction step it is no longer known which aspects actually motivated a user to bookmark an item or who assigned which tags to an item. In the next chapter, we will try to further reduce information loss by following a tensor-based approach to recommender design. The presented method uses item, tag, and user observations in parallel in order to better understand the intentions behind user-given tags with strong benefits for scenarios such as tag recommendation and tag-based social search.

Chapter 8

Translating tags for advanced user models

Folksonomies come in two forms, depending on the underlying tagging rights (see Section 1.2). *Narrow* folksonomies restrict the bookmarking and thus tagging of resources to only a limited number of users. Common examples of this type of folksonomy are services like Flickr or YouTube, where photos and videos are labeled by their owners. On the other hand, there exist *broad* folksonomies, such as Delicious or Bibsonomy, where items can be bookmarked and tagged by the entire community, such that the tagging activity of one user can be related to the activity of other users. The difference in tagging rights is also expected to have a severe impact on the observed motivations behind tagging. Whereas users of narrow folksonomies may try to promote their content by assigning common tags, i.e. tags other users will most likely search for, this motivation is less present in broad folksonomies, where users tag other users' content. As a result, users of broad folksonomies have a limited interest in the social aspects of tagging, but will assign tags for personal content retrieval. In consequence, we observe these users developing their own tag vocabularies (personomies) which may differ drastically from those of other users.

A user who wants to structure large collections by tags is not as free as it seems. Instead, there exists an underlying interest in a stable and proven set of category tags for more efficient content retrieval at a later stage. Accordingly, we observe that individual tagging habits remain relatively stable if long-term effects, i.e. changing user interests, can be ignored (see Section 2.3). Apparently, this trend is further strengthened by the tag recommendation functionalities that all popular tagging sites have introduced to their services. However, an individual user's interest in a converging set of category tags is not a social phenomenon. Instead, we observe that even if two users bookmark conceptually similar content, the tags they prefer for categorizing this content are likely to differ. It simply appears to be a matter of personal taste which category tags a user chooses among existing synonymous forms, singular or plural forms, abbreviations, etc. Also, we find that individual tag vocabularies often reflect the multilingualism of users or their different levels of expertise.

From an information retrieval perspective, the fact that tags are highly personalized may be ignored on a macroscopic scale, where cumulative effects dominate over local heterogeneities. In fact, our previous studies revealed that resources in broad folksonomies develop a characteristic and stable tag spectrum independent of user-level differences (see Section 2.3). Consequently, tags have been successfully used to calculate the similarity between items [LGZ08, RHMGM09, MCM+09] as well as for item clustering and concept extraction [HRS07, AYGS09, LGZ08]. However, effects due to customized tag assignments (TAS) cannot be neglected for applications and scenarios such as personalized tag-based search, tag recommendation, or user interest discovery, which focus on the needs of individual users.

In this chapter, we provide evidence that users in broad folksonomies follow different tagging strategies. We present a novel, user-centric tag model (UCTM) that overcomes the problems this diversity induces in many potential application areas, including tag recommendation, user interest discovery, and tag-based search. Our model infers the semantics of tags based on who applied them and by looking at the items they have been assigned to. By studying these contexts and their co-occurrence with the tags chosen by other users, we then derive translational mappings between individual personomy tags and the corresponding folksonomy. These mappings help us to overcome common problems related to tag ambiguity, synonymous tags, and biases towards singular or plural forms. The translation of personomy tags even allows us to account for multilingual communities. Furthermore, by translating user tags to the language of the community, we can deduce the meaning of a tag based on who assigned it and design more robust user models that overcome inter-user discrepancies in tagging. On the other hand, translating from the folksonomy language to a user's individual language enables us to predict the concepts (tags) that a user will associate with an item based on the associations of other users. We evaluate the applicability of our translational approach in two user-centric scenarios: tag recommendation and tag-based social search.

Tag translation for tag recommendation. Tag recommenders support a user during the posting process by suggesting potentially relevant tags. We consider the task of graph-based tag recommendation, where no additional information about the underlying items, tags, or users is given. Our approach allows us to translate community tags previously assigned to a resource into a user's own tagging language, producing more accurate predictions. Tag recommendation is probably the best-researched recommendation scenario in the context of folksonomies, see e.g. [JMH+07, HRGM08, GBM+09, SZL+08, SNM08, RBMNST09]. This scenario therefore allows us the evaluation of our approach with respect to previously presented methods.

Tag translation for tag-based social search. Folksonomies have become important sources for content search, since they aggregate the interests of thousands or millions of users and index content by tags. Our second application scenario considers a user who searches for interesting items related to a tag from her personomy. In this use case, we assume that the user wants to find relevant content related to topics she has been interested in before. Our goal is to simplify this task. Instead of having to guess how other users labeled relevant content, our solution allows the user to rely solely on her own tag vocabulary for search. The challenges in this scenario are not only the personalized retrieval of interesting items, but also the identification of the individual search intention behind a given tag. We will show that this intention can be inferred by our translation approach, yielding more useful search results. Although we limit our discussion here to a search scenario, we believe that our findings also apply to related tasks such as the topic-aware recommendation of items or the topic-aware detection and indication of trends.

The remainder of this chapter is structured as follows. After a short introduction of the utilized datasets in Section 8.1, we first discuss the motivations our translational approach is built upon and then verify the assumptions behind these motivations by analyzing the data of two real-world folksonomy (Section 8.2). We then present our translational model in Section 8.3 and evaluate its accuracy in tag recommendation and tag-based social search scenarios (Sections 8.4 and 8.5).

8.1 Datasets

We conduct our analysis and evaluate our translational model on the Delicious and Bibsonomy datasets we already used in the previous Chapter. Both folksonomies were extensively studied with respect to the problem of tag recommendation [HJSS06a, JMH+07, RBMNST09, SNM08]. Similar data from

Bibsonomy was also used during the 2008 and 2009 ECML PKDD Discovery challenges on tag recommendation[1][2], resulting in a variety of proposed solutions [RST09, KF09, MBA+09, TSD08, Lip08, LYHM09]. This gives us the opportunity to compare the results of our own tag recommendation model to previously presented achievements. On the other hand, studies on tag-based social search in a folksonomy environment are rather scarce, e.g. [BXW+07, WZY06], and all existing publications use different datasets for their evaluations.

8.2 The need for tag translation

Our translational approach is motivated by several assumptions. This section examines the validity of these assumptions by analyzing the available data.

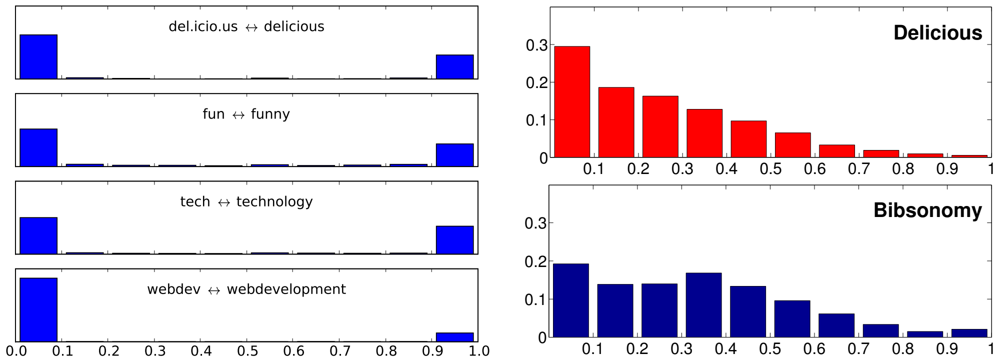
A1: People tag for personal later retrieval

The intention to save content for possible later use is a general characteristic of folksonomies [Wal04]. To reach an optimal retrieval efficiency, folksonomy members will tend to avoid the usage of synonymous tags. Furthermore, one might expect that folksonomy users are less interested in the social effects of their tagging behavior, resulting in individual tagging strategies.

To exemplify this behavior, Figure 8.1(a) investigates to what degree users of our Delicious dataset mix synonymous tags, such as “webdev” versus “webdevelopment”. The x-axis shows the ratio at which a user selects one of the two synonyms. The figure illustrates that users tend to avoid using both tags in parallel, but rather stick to their initially chosen alternative. Similar observations can be made for Bibsonomy. Table 8.1 lists the tags which two Bibsonomy users assigned to the same resources as well as the Top 5 community tags. Both users have developed standardized sets of category tags, avoiding the mixture of equivalent tags, such as “collaborative” versus “collaboration” or “folksonomy” versus “folksonomies”. These categories do not necessarily have to overlap with the community opinion, but may satisfy more personal classification needs, as demonstrated by *User237*.

Further support for our assumption comes from the observation that 15 percent of all Delicious users in our dataset group tags to so called *tag bundles*. This feature of Delicious lets users define categories and list the personomy tags that should fall into each category, providing users with an additional

[1] http://www.kde.cs.uni-kassel.de/ws/dc09
[2] http://www.kde.cs.uni-kassel.de/ws/rsdc08/



(a) (b)

Figure 8.1: (a) Intra-personomy distribution of synonym tags in Delicious. The x-axis shows the ratio of the right tag w.r.t the occurrence of both tags in each user's personomy. Users tend to avoid to mix equivalent tags for more efficient content categorization. (b) Cosine similarities between the tags assigned to the same resources by two randomly selected users. Only user pairs that shared at least 5 items have been included in the evaluation. The tags which users apply to the same resources do not necessarily overlap. In consequence, users that obviously share the same interests by bookmarking the same content do not have to appear similar in the tag space.

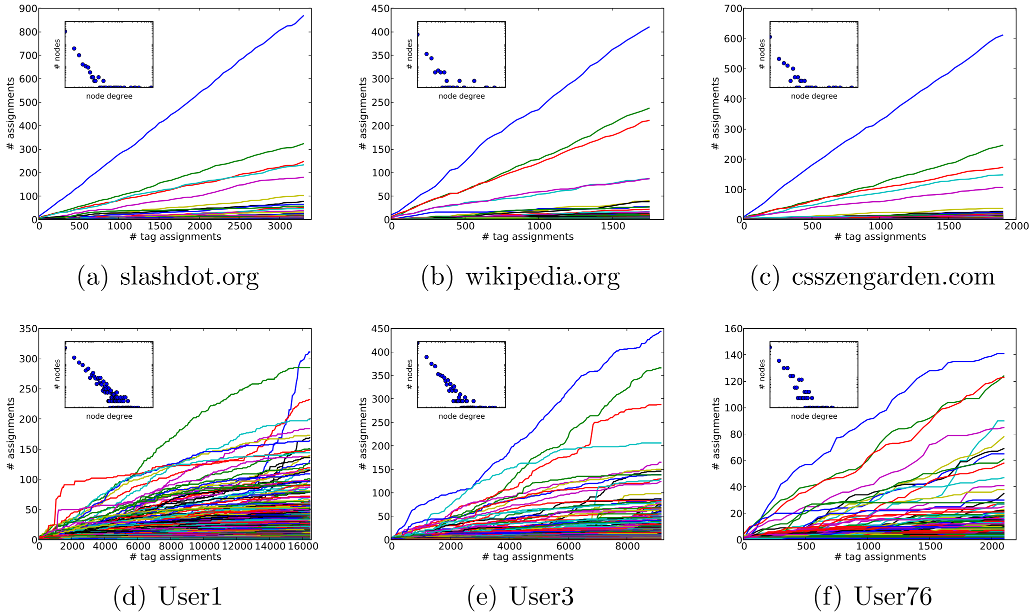
top-down classification technique (see Section 2.3). The frequent use of this technique further highlights the retrieval motivation behind tagging.

A2: People tag differently

Table 8.1 also highlights inter-user differences in tagging strategies. While *User107* has a tendency towards commonly assigned tags, we find the second user (*User237*) to have developed a personalized set of labels including tags, such as "diplomathesis" or "closelyrelated", that rather describe the user's relation to the respective content than the content itself. Figure 8.1(b) further emphasizes the user-centric nature of tagging by looking at the cosine similarity between the tags two randomly selected users assigned to the same resources. The evaluation only includes pairs of users who share at least five items. Independent of the dataset, about 20 to 30 percent of all pairs of users labeled the same resources in such a different manner that there exists nearly no overlap between the assigned tags. The divergence of categorization strategies likely exceeds the one shown in Figure 8.1(b), because, as we have seen in Chapter 2, many overlapping tags describe a specific

Table 8.1: Tags assigned to publications by the community (Top 5) and two exemplary users (Bibsonomy).

Publication title	**Community**	**User107**	**User237**
Ontologies Are Us: A Unified Model of Social Networks and Semantics	social, ontology, ontologies, folksonomy, socialnetworks	social, ontology, network, sna	socialnetworks, ontologies, diplomathesis, closelyrelated
Harvesting social knowledge from folksonomies	folksonomy, tagging, social, knowledge, folksonomies	social, tagging, knowledge, folksonomy, collaborative	diplomathesis, closelyrelated
Towards the Semantic Web: Collaborative Tag Suggestions	tagging, folksonomy, web20, collaborative, recommender	tagging, collaborative	tagging, folksonomy, diplomathesis, closelyrelated
The PageRank Citation Ranking: Bringing Order to the Web	pagerank, search, ranking, ir, web	search, ir, ranking, pagerank	diplomathesis, eventuallyuseful
Automated Tag Clustering: Improving search and exploration in the tag space	clustering, tagging, search, folksonomy, toread	folksonomy, clustering, tags, collaborative, search, filtering, exploration, tag	diplomathesis, closelyrelated
Integrating Folksonomies with the Semantic Web	folksonomy, semanticweb, tagging, semantic, clustering	semanticweb, tagging, folksonomy, projet	toread
The Dynamics and Semantics of Collaborative Tagging	tagging, semantics, ontology, folksonomy, dynamics	tagging, collaborative, filtering	ontology, closelyrelated, diplomathesis, semantics, collaborativetagging
Exploring social annotations for the semantic web	social, tagging, folksonomy, semanticweb, annotation	semanticweb, social, annotation	diplomathesis, folksonomy, semanticweb, tagging, closelyrelated
Collaborative tagging as a tripartite network	tagging, folksonomy, network, collaboration, clustering	tagging, network, collaborative, projet, projtags	diplomathesis, folksonomybackground
Folksonomy as a Complex Network	folksonomy, tagging, network, folksonomybackground, social	network, complex, folksonomy	diplomathesis, folksonomybackground



(a) slashdot.org (b) wikipedia.org (c) csszengarden.com

(d) User1 (e) User3 (f) User76

Figure 8.2: Tag distribution over time for three Delicious URLs and users. Each line corresponds to the frequency of one tag w.r.t the overall number of tag assignments (TAS). The small plots show the corresponding node degree distributions (*log-log-scale*). Resources develop a characteristic and stable tag spectrum strongly dominated by very few tags. User vocabularies are more dynamic and react to shifts in a user's interests or tagging behavior. The node degree distributions of personomies exhibit power-law characteristics, whereas the contrast between very frequent and very rare item tags is even above an expected power-law divergence. See Section 2.3 for details.

item and are not part of any categorization scheme. Therefore, even in cases where two users obviously share common interests by bookmarking the same content, they do not necessarily appear similar in the tag space. Deriving user similarities by simple personomy comparison [ZC08, SGMB08] will hence lead to suboptimal results. We therefore propose first to translate personomies into a unified global representation before taking further steps.

A3: People share a common understanding about content

A prerequisite of our model is the relative stability of tag distributions from an item perspective. Figures 8.2(a)-(c) depict the tag distributions of three

URLS within the small Delicious corpus. Even though based on less data, the plots show exactly the same stable vocabularies we previously reported for the full corpus in Chapter 2. All plots show a convergence after only a few bookmarking events. The collaborative labeling of content thus leads to the emergence of a characteristic, stable tag spectrum despite of user-level discrepancies. The stability of the spectra justifies our claim of translating to the folksonomy vocabulary even if an item has been bookmarked by only a subset of users.

A4: User interests are stable in the short-term

Our translation approach implicitly assumes that the user models which have been trained on past data are valid for the prediction of future user behavior. This makes it essential to look at how fast user interests and tagging strategies, that both manifest within personomies, actually change. Figures 8.2(d)-(e) support previous results from Chapter 2 that demonstrate the relative stability of intra-personomy tag distributions. However, these appear much less static than for resources reflecting shifts in user interests or changing labeling strategies. Consequently, user tag models, even though representative in the short term, are likely to require constant updates. Furthermore, comparing the tag distributions of URLs and users unveils that personomies are less dominated by very frequent tags than resources are. The deviating distributional patterns between personomy and item tags make direct calculations of user item similarities, e.g. by calculating the cosine similarity between personomies and item vocalbularies [NM07, CBC08], less effective. Our approach overcomes these problems by mapping user tags to item derived vocabularies before calculating similarities.

Another way to investigate the stability of personomies is to look at the number of newly assigned tags that have not been in the vocabulary before. As reported in Chapter 2 the percentage of new tags covered by a user's personomy constantly increases as the personomy grows. This is important for the tag recommendation scenario, since the majority of new tags can be predicted from the tags of a user's personomy or an item's vocabulary.

The assumptions explained in this section are the core building blocks of our translation approach which we describe in the following section.

8.3 Translating personomies

We assume that each user has a distinctive vocabulary of tags that can be translated to the folksonomy vocabulary by looking at the tag co-occurrences within the shared item space. Applying tensor calculus[3], we can construct the tensor $\boldsymbol{\mathcal{T}}_{T\tilde{T}U}$ that contains these (item-normalized) tag co-occurrence counts for each user as

$$\boldsymbol{\mathcal{T}}_{T\tilde{T}U} = \boldsymbol{\mathcal{Y}}_{ITU} \times_I \mathbf{A'}_{TI}, \tag{8.1}$$

where the T and $\tilde{T}$ modes of the translation tensor represent the folksonomy and personomy tag dimensions, $\mathbf{A'}_{TI}$ is the transposed and item-normalized stochastic version of the global item-tag co-occurrence matrix $\mathbf{A}_{IT}$, and $\times_X$ denotes the matrix product with a tensor in the mode corresponding to the set X. The value of each element in $\boldsymbol{\mathcal{T}}_{T\tilde{T}U}$ is thus equal to

$$\boldsymbol{\mathcal{T}}_{t\tilde{t}u} = \sum_{i \in I} \boldsymbol{\mathcal{Y}}_{i\tilde{t}u} \mathbf{A'}_{ti}. \tag{8.2}$$

Since $\boldsymbol{\mathcal{Y}}_{i\tilde{t}u}$ will be 1 if $(i, \tilde{t}, u) \in Y$ and 0 otherwise, this can also be written as

$$\boldsymbol{\mathcal{T}}_{t\tilde{t}u} = \sum_{i \in I_{u\tilde{t}}} \mathbf{A'}_{ti}. \tag{8.3}$$

Each tag $\tilde{t}$ from the personomy of a user u is thus represented by a tag vector $\boldsymbol{\mathcal{T}}_{T\tilde{t}u}$ that aggregates the tag distributions of all items that user u has labeled with $\tilde{t}$. One important aspect of our model is that it considers the user as a part of the folksonomy she is mapped to. This allows us to derive primitive mappings even in cases where a user does not share any items with other users or where the shared items are sparsely tagged.

Based on $\boldsymbol{\mathcal{T}}_{T\tilde{T}U}$, tags from a user's personomy represented by $\tilde{\mathbf{t}}$ can now be translated to the folksonomy vocabulary by simple vector multiplication:

$$\mathbf{t}(\tilde{\mathbf{t}}, u) = \frac{\boldsymbol{\mathcal{T}}_{T\tilde{T}u} \overline{\times}_{\tilde{T}} \tilde{\mathbf{t}}}{|\boldsymbol{\mathcal{T}}_{T\tilde{T}u} \overline{\times}_{\tilde{T}} \tilde{\mathbf{t}}|}, \tag{8.4}$$

where $|.|$ represents the sum of all vector elements. Each value of $\mathbf{t}(\tilde{\mathbf{t}}, u)$ gives us a weight for the user meaning t when saying $\tilde{\mathbf{t}}$. Analogously, we can translate from the global tag language to a user's personomy:

$$\tilde{\mathbf{t}}(\mathbf{t}, u) = \frac{\boldsymbol{\mathcal{T}}_{T\tilde{T}u} \overline{\times}_T \mathbf{t}}{|\boldsymbol{\mathcal{T}}_{T\tilde{T}u} \overline{\times}_T \mathbf{t}|}. \tag{8.5}$$

We will later use the mapping from Equation 8.5 to identify the most likely user tags for an item given the tags previously assigned by the community. The translation of user tags to the global tag vocabulary from Equation 8.4, on the other hand, will help us to infer the meaning of a query tag in the search scenario.

[3] For details on notation, set definitions, and an introduction of tensors see Section 1.4.

8.4 Scenario 1: Tag recommendation

Tag recommendation within a folksonomy is the task of predicting $\boldsymbol{\mathcal{Y}}_{iTu}$ for a previously unseen bookmark, i.e. $(i, u) \notin B$.

The tag recommender

We approach the problem of tag recommendation by translating the tag spectrum of the item i to the personomy of user u. The weights $\widehat{\mathbf{t}}(i, u)$ for each personomy tag are calculated using Equation 8.5 as

$$\tilde{\mathbf{t}}(i, u) = \frac{\boldsymbol{\mathcal{T}}_{T\tilde{T}u} \overline{\times}_T \mathbf{A'}_{iT}}{|\boldsymbol{\mathcal{T}}_{T\tilde{T}u} \overline{\times}_T \mathbf{A'}_{iT}|}, \tag{8.6}$$

where $\mathbf{A'}_{iT} = \mathbf{A'}_{IT} \overline{\times}_I \ \mathbf{i}$ denotes the item's community tag distribution. Based on $\tilde{\mathbf{t}}(i, u)$ we can now predict the personomy tags a user will assign to a new item. However, $\tilde{\mathbf{t}}(i, u)$ only contains weights for tags within the user's personomy, whereas all tags the user did not assign yet will have zero weights. Recommending tags solely based on $\tilde{\mathbf{t}}(i, u)$ would thus disregard the previously noted fact that many tags are item specific (see also Figure 8.2). We therefore extend our model to include item tags to

$$\widehat{\mathbf{t}}(i, u) = \alpha \, \widetilde{\mathbf{t}}(i, u) + (1 - \alpha) \, \mathbf{t}(i), \tag{8.7}$$

with $\mathbf{t}(i) = \mathbf{A'}_{iT}$. Tags are thus recommended from a mixture of personomy and item tags, where the parameter $0 \leq \alpha \leq 1$ controls the mixture ratio. For each user item combination, we then recommend the N tags with the highest value in $\widehat{\mathbf{t}}(i, u)$.

Experiments

We run our evaluations on the p-cored versions of the Delicious and Bibsonomy datasets from the last chapter. From both datasets we randomly select 10 percent of all bookmarks for testing and keep the rest for training. For each test bookmark, we then predict a ranked list of tags based on $\widehat{\mathbf{t}}(i, u)$ and calculate the achieved Precision, Recall and the corresponding F_1 measures at different ranks N. We report results averaged over all bookmarks and 10 test runs. The best value of the α parameter is determined using a *brute-force* approach. We choose the α value that performs best in terms of maximal reached F_1 measure.

Other recommenders

We compare the results achieved by our UCTM approach to a variety of baseline recommenders.

MostPopular This recommender weights tags by their global occurrence in the training data.

MostPopular$_I$, MostPopular$_U$ These recommenders weight tags by their co-occurrence with the given item or user.

MostPopular$_{I,U}$ This recommender weights tags by a mixture of their item and user co-occurrence distributions:

$$\widehat{\mathbf{t}}(i,u) = \beta\frac{\mathbf{t}(i)}{|\mathbf{t}(i)|} + (1-\beta)\frac{\mathbf{t}(u)}{|\mathbf{t}(u)|},$$

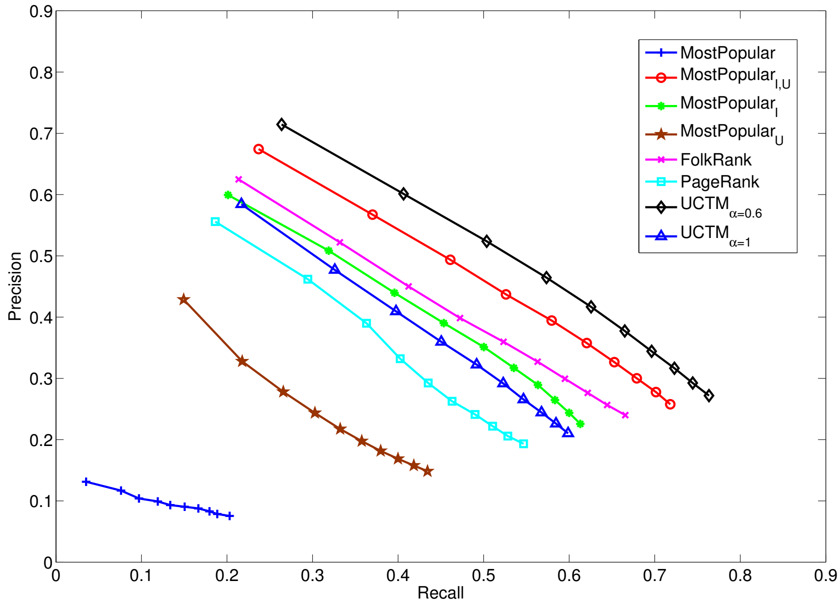
with $0 \leq \beta \leq 1$. A similar recommender was reported to perform reasonably well in [JMH+07].

Adapted PageRank and FolkRank Both recommenders are identical to the ones used in Chapter 7. The preference vector $\mathbf{p}$ is set to 1 except for the entries that correspond to the input item and user and that are set to $|I|$ and $|U|$ respectively. The parameter d is again set to 0.7. This resembles the setting in [JMH+07].

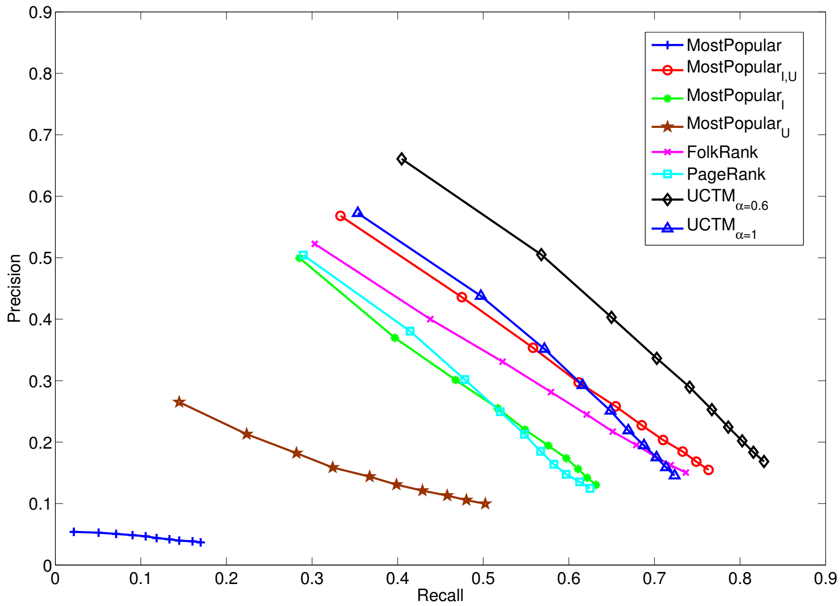
The optimal β parameter of the $MostPopular_{I,U}$ model is again chosen using a *brute-force* approach. For all recommenders we always suggest the highest weighted N tags and evaluate the performance at rank N.

Results

Figure 8.3 presents the Precision and Recall values of all recommenders when suggesting up to 10 tags. Independent of the dataset, we observe very similar results, with our $UCTM$ method performing best. The importance of tag translation is highlighted when comparing the results of the $MostPopular_U$ with the $UCTM_{\alpha=1}$ recommender. Both models only return the tags a user has already used before. However, selecting tags based on the (translated) community opinion about an item is much more accurate, since it respects the item's nature. The comparably good result of the $MostPopular_I$ recommender is most likely a consequence of the previously discussed very high density of item tags, which simplifies the prediction of popular tags. However, combining item and (translated) user tags still yields large performance



(a) Bibsonomy



(b) Delicious

Figure 8.3: Precision/Recall curves for various tag recommenders on two datasets. The user-centric tag model (UCTM) improves prediction precision independent of the dataset.

gains, and a $MostPopular_{I,U}$ recommender with β set to 0.5 produces reasonable good results on both datasets, outperforming even more advanced approaches such as the $FolkRank$ method.

The best recommendations on both datasets are given by a $UCTM$ recommender with a mixture ratio of $\alpha = 0.6$. This recommender reaches a maximal F_1 measure of 0.514 on Bibsonomy and 0.533 on Delicious, performing significantly better than the $MostPopular_{I,U}$ (0.478/0.448) and the FolkRank (0.430/0.413) methods. The performance gain of our approach on the Delicious dataset is probably a consequence of larger personomy sizes that result in more robust translational models. The very good performance of our $UCTM$ approach underlines the potential of tag translation for the task of tag recommendation. This potential is even higher if we consider that our method of fusing translated user and item tag models in Equation 8.7 is rather basic and still leaves much space for improvements.

8.5 Scenario 2: Social search

In the social search scenario we consider a user who wants to find content related to one of the tags from her personomy. Searching for items will generally require that the user guesses the tags other users might have assigned to relevant content. Instead, we enable the user to search based on her individual vocabulary and translate the user query to the global vocabulary. The proposed social search service thus supports users in discovering new content with respect to the topics they have previously shown interest in. For instance, *User237* from Table 8.1 may search for publications using the tag "diplomathesis" and expecting results similar to the documents she previously labeled with this tag. This allows her to discover content related to her own work of which she was previously unaware of. The challenge in this scenario is to identify the search intention that a user associates with the given tag. From a technical perspective, we are required to predict $\boldsymbol{\mathcal{Y}}_{Itu}$, where $(i, u) \notin B$ and $t \in T_u$.

Translating query tags

We divide the search process into two steps: The translation of the query tag to the global vocabulary and the actual item retrieval. Translating a single user tag $\tilde{t}$ only requires to look at it's previous co-occurrence with other tags represented by the vector $\boldsymbol{\mathcal{T}}_{T\tilde{t}u}$ within the translation tensor $\boldsymbol{\mathcal{T}}_{T\tilde{T}U}$. In the second step, we then identify the items which generally fall together with these community tags calculating the weight vector $\widehat{\mathbf{i}}(\tilde{t}, u)$ as

$$\widehat{\mathbf{i}}(\tilde{t}, u) = \frac{\boldsymbol{\mathcal{T}}_{T\tilde{t}u} \times_T \mathbf{A^*}_{IT}}{|\boldsymbol{\mathcal{T}}_{T\tilde{t}u} \times_T \mathbf{A^*}_{IT}|}, \tag{8.8}$$

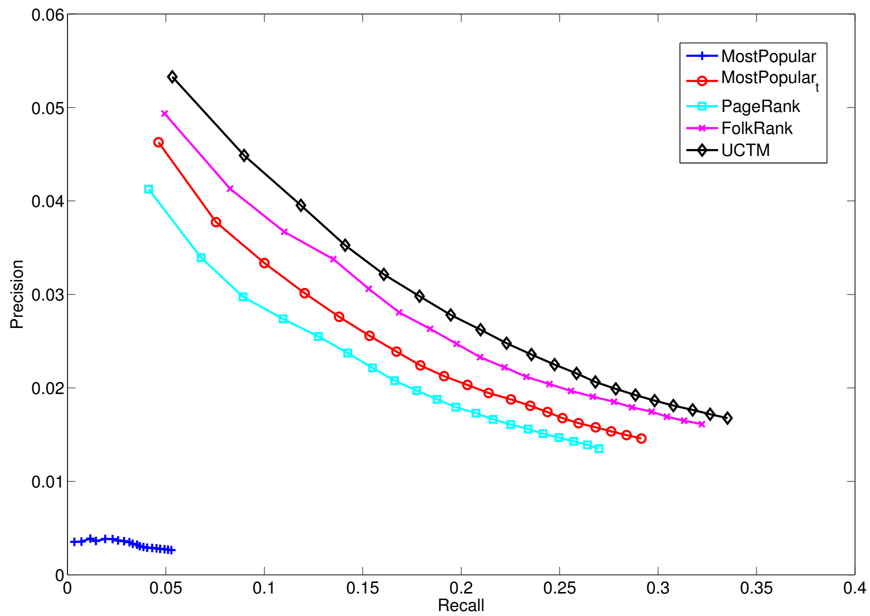
where $\mathbf{A^*}_{IT}$ is the tag-normalized stochastic version of the matrix $\mathbf{A}_{IT}$. In this work, we only consider single-tag search queries. However, our model can be easily extended to multi-tag queries by replacing the $\boldsymbol{\mathcal{T}}_{T\tilde{t}u}$ of Equation 8.8 with $\boldsymbol{\mathcal{T}}_{T\tilde{T}u} \overline{\times}_{\tilde{T}} \tilde{\mathbf{t}}$, where $\tilde{\mathbf{t}}$ is the tag vector corresponding to the query. We set all values of $\widehat{\mathbf{i}}(\tilde{t}, u)$ to 0 where $(i, u) \in B$, since items are bookmarked only once per user. We then recommend the N items with highest weight in $\widehat{\mathbf{i}}(\tilde{t}, u)$.

Experiments

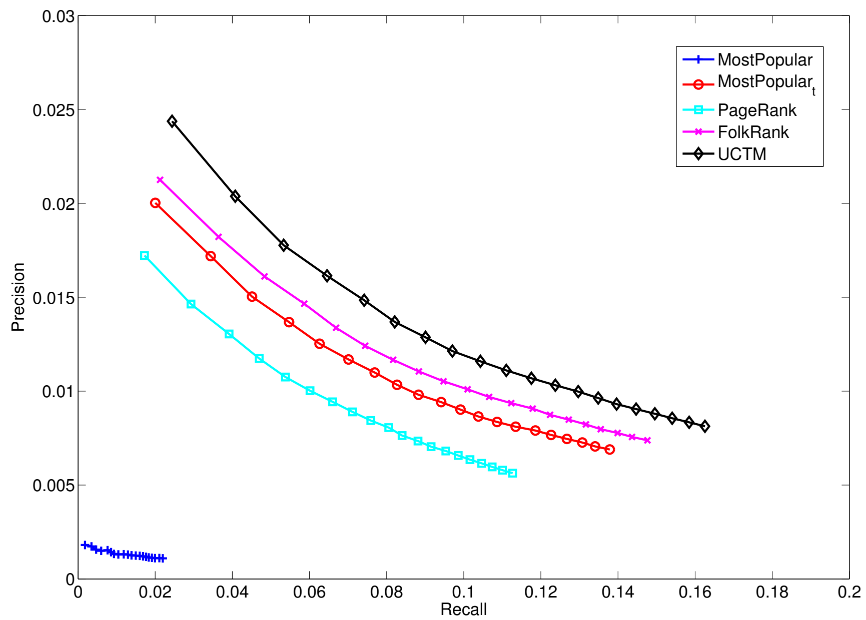
We apply the same test setup as in the tag recommendation scenario. For each bookmark of the test set we now randomly select a tag to simulate a query and try to predict the item the user had labeled with this tag. To avoid item specific tags with little expressiveness for user interests, we require that each query tag was applied at least 5 times by the user in the training data. For each user tag combination we then recommend the N items with highest weight in $\widehat{\mathbf{i}}(\tilde{t}, u)$. As for the tag recommendation scenario, we test our approach against a variety of other methods, including the previously discussed **Adapted PageRank** and **FolkRank** algorithms. Furthermore, we report results for a **MostPopular** item recommender as well as for a tag-aware item recommender (**MostPopular**$_T$) that ranks items according to their previous co-occurrence with the query tag, but does not perform any personalization.

Results

The Recall and Precision values for the first 20 ranks and for all search methods are displayed in Figure 8.4. Both plots demonstrate a superior result quality for our translation approach ($UCTM$). The query tag translation improves the *Recall@10* by about 15 percent on Bibsonomy and 18 percent on Delicious compared to the $MostPopular_T$ baseline method. Our translational model also performs better than the $FolkRank$ algorithm, yielding *Recall@10* improvements of 7.5 percent and 10 percent. This becomes even more meaningful if we consider that the $FolkRank$ algorithm incorporates collaborative filtering information, e.g. information about users with similar bookmarking patterns, whereas our approach personalizes solely by trying to



(a) Bibsonomy



(b) Delicious

Figure 8.4: Precision/Recall curves for the social search task on two datasets. Both plots show an improved performance when translating user tags based on the UCTM model.

understand the user's search intention. Similar to the tag recommendation task, we find the *AdaptedPageRank* algorithm to perform worse than simple baseline methods.

The prediction task in this search setup reoccurs in other application scenarios, such as the topic-aware recommendation of items or the topic-aware detection of new content. We believe that our translation approach can also help to identify user interests in these scenarios.

8.6 Chapter conclusions

This chapter investigated the nature of tagging within folksonomies from a user-centric perspective. In our studies on two real-world folksonomies, we discovered that users who tag for content categorization develop distinct tag vocabularies over time. This heterogeneity vanishes when the tags of many users are aggregated, resulting in characteristic tag distributions for resources. However, the personal aspect of tagging cannot be ignored in user-centric scenarios, such as tag recommendation or tag-based social search.

We presented a novel approach to tag translation that maps user tags to the global folksonomy vocabulary, using the labeled resources as intermediates. Based on these mappings we were able to infer the meaning of a user tag and to accurately predict which tags a user would assign to new content. We evaluated the applicability of our approach in two relevant use cases, tag recommendation and social search. Our experimental results showed that tag translation improves prediction accuracy in both scenarios, and our user-centric tag model produced more accurate recommendations then existing *state-of-the-art* algorithms.

One shortcoming of our translational approach is the missing incorporation of temporal aspects, such as changing tagging behaviors or shifts in user interests. This may result in suboptimal tagging models, which in turn reduces the quality of potential recommenders. In the next chapter, we will look exclusively at temporal patterns within the global tagging behavior of a folksonomy, and present a tag-based trend detection method which recommends upcoming topics.

Chapter 9

Tag-based trend detection

In all recommendation scenarios discussed so far we followed common practice and ignored the potential impact of temporal aspects when building our models as well as during evaluation. This is of course a simplification, since time can strongly influence the quality of recommendations and the validity of the trained models. Furthermore, since folksonomies are open in nature, there is a continuous stream of new items, users, and tags which leads to constantly growing object sets. Recommenders that ignore temporal aspects will generally underestimate the relevance of new objects, since these tend to be less connected within the folksonomy graph.

The impact of temporal aspects on folksonomy recommendation has been mainly ignored in the past. As a result, there exist many potential research directions within this area that are still not sufficiently covered by publications. This comes to a surprise, since the importance of the factor time is well known from other recommendation settings, e.g. see [Kor09] for the importance of temporal aspects in movie recommendation. From all choices, this chapter tackles the problem of tag-based trend detection in folksonomies. Trend detection can help users to discover rising topics within a folksonomy at an early stage. Trend detection can also create new opportunities in areas, such as product tracking or marketing in on-line communities. So far, existing studies mainly focus on the identification of opinion leaders and the detection of trends in the blogosphere or on the web [ALTY08, CTT06, ACH+04]. Folksonomies have only recently been discovered as potential sources for trend detection [DKM+06, HJSS06b]. We will discuss these works in detail when comparing our approach to existing methods in Section 9.5.

In this chapter, we present a novel probabilistic method for the detection of trends within the tagging behavior of a folksonomy. Our approach considers trends as statistical anomalies and weights trend patterns based on the unlikelihood of their appearance. We find that our model detects upcoming as well as periodic tag trends within a large-scale Delicious corpus. Comparative studies show that our approach is more robust than previously proposed measures. Another contribution of our work is the integration of the *diffusion-of-attention* concept from Chapter 4 into the trend detection model. This allows us to move from an occurrence-based perspective to a change-based one. By counting only the number of new users reached by a tag, we are able to drastically limit the impact of otherwise influential spam users without loosing the descriptiveness of our model. Although we limit our discussion to tag-based trends, we believe that the sound statistical foundation of our approach guarantees its applicability to any type of count-based time series.

This chapter is structured as follows: We start by describing the data set on which we perform our evaluations. We then apply the *diffusion-of-attention* concept from Chapter 4 in order to limit the impact of spam on our detection results. Section 9.3 introduces our probabilistic trend detection model, which we evaluate in Section 9.4. The chapter ends with a comparison of our approach to other trend detection measures in Section 9.5.

9.1 The dataset

We run our experiments on a subset of the Delicious corpus described in Section 1.5. From the original corpus we only consider $400,000$ users. This reduction is owed to the fact that the experiments were run before the crawling of the full Delicious corpus ended. The evaluation dataset therefore represents the part of the crawl that was retrieved first. Because our crawling method downloaded frequent users first, this snapshot is nearly identical to the data one would get when reducing the full corpus to the $400,000$ most active users. The reduced dataset contains $105,258,294$ bookmarks and $331,449,289$ tag assignments.

9.2 Avoiding spam

It seems common practice to measure the importance of a tag by its node degree in the folksonomy graph G which is equivalent to the overall number of tag assignments that contain this tag [DKM+06, GH06, MNBD06]. However, this does not take into account that most collaborative tagging systems are vulnerable to spam. As previously noted, this spam strongly distorts the natural data patterns and therefore needs to be taken care of before any meaningful trends can be detected. In order to limit the impact of spam, we therefore apply the *diffusion-of-attention* concept described in Chapter 4. The impact of a tag is hence not measured by its overall occurrence, i.e. the number of assignments that contain it, but by the number of users who have assigned it. Thus, each user can only contribute the same amount of tag popularity, limiting the potential impact of very active spam users. From a technical point of view, moving from an occurrence to an attention concept is equivalent to the conversion of the frequency matrix $\mathbf{A}_{TU}$ to a binary matrix $\tilde{\mathbf{A}}_{TU}$, where $\tilde{\mathbf{A}}_{tu} = 1$ if $\mathbf{A}_{tu} > 0$, and $\tilde{\mathbf{A}}_{tu} = 0$ otherwise.

Since we investigate temporal tag developments, we are interested in the number of new users who assign a tag for the first time between the timestamps δ_1 and δ_2. All new edges between users and tags during this period are represented by the non-zero entries of a matrix $\Delta\tilde{\mathbf{A}}_{TU}$ calculated as

$$\Delta\tilde{\mathbf{A}}_{TU} = \tilde{\mathbf{A}}_{TU}(\delta_2) - \tilde{\mathbf{A}}_{TU}(\delta_1).$$

The increase of attention for a tag t, which represents the number of users that have assigned t for the first time between δ_1 and δ_2, is then simply

$$f(t) = \sum_{u \in U} \Delta\tilde{\mathbf{A}}_{tu}.$$

Considering only first-time tag assignments and disregarding all later user-tag combinations reduces the number of total tag assignments within our corpus from 331 million to 70 million.

9.3 A probabilistic model for trend detection

We consider an object, i.e. a tag, to be a trend if it attracts the attention of significantly more new users in the current time frame than it did in the previous one. Since we treat trends as statistical anomalies, we measure their significance using a probabilistic generative model. For this purpose, we compare the tag distributions $\mathbf{t}_0$ and $\mathbf{t}_1$ that result from all frequency counts $f_0(t)$ and $f_1(t)$ over two consecutive periods Δ_0 and Δ_1. We then assume that all tag events from $\mathbf{t}_1$ are generated independently at random from $\mathbf{t}_0$. The probability of observing a tag t estimated over Δ_0 is given by $p_0(t) = f_0(t)/\sum_{t' \in T} f_0(t')$. The likelihood of observing t at a frequency $f_1(t)$ in period Δ_1 can then be calculated as

$$p(f_1(t)) = \binom{n_1}{f_1(t)} p_0(t)^{f_1(t)} (1 - p_0(t))^{n_1 - f_1(t)}, \tag{9.1}$$

where $n_1 = \sum_{t \in T} f_1(t)$ denotes the number of all observations in Δ_1. However, this simple model will fail in cases of previously unobserved tags. Therefore, in order to handle situations where $f_0(t) = 0$ and $f_1(t) > 0$, we assume a beta distribution as prior

$$F(p; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}, \tag{9.2}$$

with $\alpha = \epsilon$ and $\beta = (|T| - 1)\epsilon$. Here, $|T|$ denotes the number of known tags by the time of measurement, and ϵ denotes the assumed prior frequency of each tag. Combining the prior and the observations in Δ_0, the $p_0(t)$ in Equation 9.1 evolves into

$$p_0(t) = \frac{f_0(t) + \epsilon}{n_0 + \epsilon|T|}. \tag{9.3}$$

Adding the prior hence has the same effect as *additive smoothing* which is well known from language modeling [CG96]. Moreover, the additive parameter ϵ allows to control the influence of the prior distribution with respect to the observations in Δ_0 and thus for adjusting the degree of smoothing. As a result, we are able to calibrate the trade-off between the detection of absolute and relative changes.

Smoothing $\mathbf{t}_0$ will result in a bias towards rare tags. This bias will appear as a trend by itself when we compare $\mathbf{t}_0$ to a non-smoothed distribution $\mathbf{t}_1$. To neutralize this effect, we also smooth the tag distribution $\mathbf{t}_1$, using the same smoothing factor ϵ. Finally, we calculate a score for all tags for which $p_1(t) > p_0(t)$ as

$$score(i) = -log(p(f_1(t))). \tag{9.4}$$

The tags with the highest scores are then considered the most significant trends. Since the exact calculation of $p(f_1(t))$ is computationally expensive, we estimate the binomial probability mass function by a Gaussian of the type $\mathcal{N}(n_1 p_0(t), n_1 p_0(t)(1 - p_0(t)))$.

9.4 Experiments

In our experimental evaluation, we generate monthly snapshots of the Delicious dataset. For each month we then determine the matrix $\Delta\tilde{\mathbf{A}}_{TU}$, from which we calculate the number of newly attracted users per tag t as $f(t) = \sum_{u \in U} \Delta\tilde{\mathbf{A}}_{tu}$. The *trendiness* of each tag within each month with respect to the previous one is calculated using the score function from Equation 9.4.

Table 9.1: Top 5 tag trends for different months and different values of ϵ. Higher values of the smoothing parameter ϵ prioritize absolute popularity changes and favor the detection of seasonal trends, such as "christmas" and "halloween". Small values of ϵ instead tend to favor new, previously unknown concepts.

		$\epsilon = 0.2$		$\epsilon = 1$		$\epsilon = 100$	
		tag	score (f_0/f_1)	tag	score (f_0/f_1)	tag	score (f_0/f_1)
Oct'07	1	inrainbows	51747 (0/148)	leopard	65453 (102/3203)	leopard	44647 (102/3203)
	2	leopard	44284 (102/3203)	radiohead	12617 (81/1231)	radiohead	7483 (81/1231)
	3	bit200f07	42420 (0/134)	opensocial	11477 (5/422)	halloween	7168 (476/2380)
	4	twine	31743 (1/285)	twine	9750 (1/285)	imap	5596 (207/1459)
	5	decenturl	23626 (0/100)	prism	9307 (27/663)	prism	3425 (27/663)
Nov'07	1	android	290314 (22/3556)	android	360132 (22/3556)	android	102030 (22/3556)
	2	kindle	130602 (3/930)	kindle	78912 (3/903)	opensocial	18865 (422/3366)
	3	gos	78083 (2/579)	gos	37159 (2/579)	kindle	8470 (3/903)
	4	gracenote	47711 (0/136)	opensocial	17378 (422/3366)	thanksgiving	5300 (158/1223)
	5	dalvik	20434 (0/89)	quicklook	9113 (14/480)	vector	4945 (1818/4030)
Dec'07	1	simpledb	1955161 (0/826)	simpledb	117574 (0/826)	simpledb	8127 (0/826)
	2	knol	937602 (0/572)	knol	56484 (0/572)	christmas	6180 (2219/4648)
	3	remastersys	39233 (0/117)	alphabetize	5078 (2/203)	2007	5314 (1030/2803)
	4	blazeds	20708 (0/85)	alphabetizer	4404 (2/189)	wii	4376 (993/2531)
	5	diso	19745 (0/83)	christmas	4353 (2219/4648)	knol	4080 (0/572)

Table 9.1 presents the Top 5 tag trends for 3 exemplary months of 2007 that are detected at different levels of smoothing. The table shows that our measure successfully detects new events, such as the announcement of Google's *Android* or of Apple's *Mac OS X Leopard*. Both trends are detected, although the tags "leopard" and "android" were already well known to the community before. These trends are so dominant that they are detected independent of the degree of smoothing. We also find our method to succeed in the detection of seasonal events, such as Halloween, Thanksgiving, or Christmas. However, detecting these seasonal trends requires a high degree of smoothing in order to tune the detector towards absolute changes in popularity. Similar patterns can be observed for other common tags, such as "2007" or "vector", which are know to the community, but then encounter a popularity boost, e.g. towards the end of the year.

Studying the impact of the ϵ parameter, we find that the tags which are identified as trends for $\epsilon = 0.2$ were likely unknown before, such that their relative development appears dramatic. On the other hand, choosing larger values for ϵ favors popular tags which are already well spread throughout the community. Furthermore, we observe that small ϵ values help discover narrow trends, i.e. trends that peak for few days and then disappear again. A higher ϵ setting will instead favor more robust trends that evolve over longer periods of time. Both these effects are emphasized in Figures 9.1 and 9.2. Figure 9.1 displays the development of the Top 5 tags from December 2007 with $\epsilon = 0.2$. Tags were generally unknown before they peak and are detected as trends. The attracted attention only lasts for few days. Smaller peaks in the aftermath of a major event are likely caused by the appearance of strongly related new items. For example, such revivals can occur when news are spread within the blogosphere, and additional blog entries discuss the same topic, e.g. the introduction of *Android*. Users that bookmark these new entries will probably use "android" as descriptive content label. Figure 9.2 displays the long term nature of the trends that are detected with $\epsilon = 100$. Only two new concepts ("knol", "simpledb") encounter enough absolute popularity gain to appear among these Top 5 trends.

9.5 Comparison to other measures

Two other publications have studied the problem of trend detection in folksonomies. In order to visualize Flickr tag trends over time, **Dubinko et al.** [DKM+06] propose an *interestingness* measure defined as

$$int(t, \Delta) = f_\Delta(t)/(C + f(t)), \tag{9.5}$$

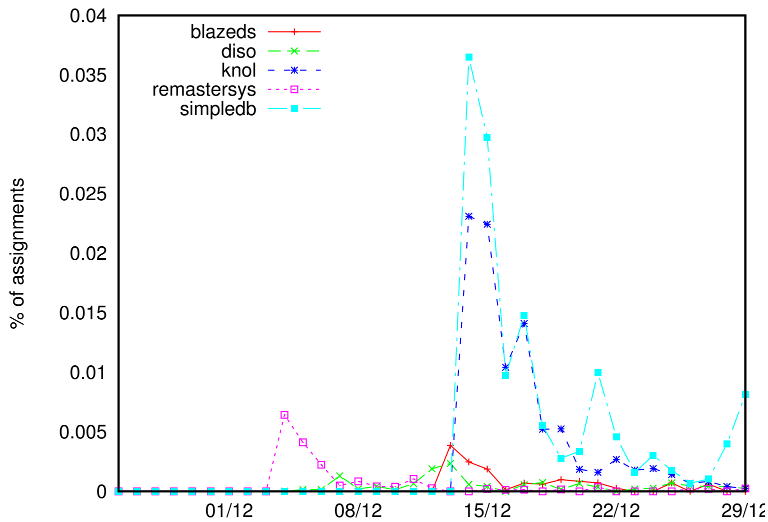


Figure 9.1: Performance over time for the Top 5 tag trends of December 2007 ($\epsilon = 0.2$). The x-axis is zoomed to December 2007. The plot displays the preference for relative improvements when choosing small values of ϵ.

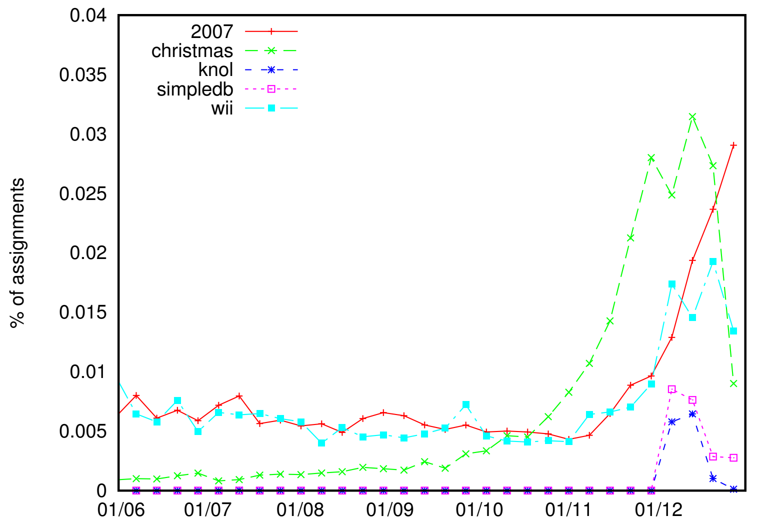


Figure 9.2: Performance over time for the Top 5 tag trends of December 2007 ($\epsilon = 100$). The x-axis is zoomed to the last 7 months of 2007. The plot emphasizes the preference for absolute changes when choosing large values of ϵ. Two tags ("2007", "christmas") represent common concepts that are present all over 2007, but peak at the end of the year. We also find the Top 2 tags from Figure 9.1 among the detected trends.

Table 9.2: Top 5 tag trends in November 2007 according to *interestingness* ($C = 50$) and *popularity change*. The detected trends by both measures do not overlap on the first five ranks.

	tag	int (f_0/f_1)	tag	pc (f_0/f_1)
1	android	0.98 (22/3556)	gravenote	2.19 (0/136)
2	kindle	0.94 (3/903)	dalvik	2.03 (0/89)
3	gos	0.92 (2/579)	23andme	2.02 (0/86)
4	hackintosh	0.89 (43/723)	blog-directories	1.97 (0/76)
5	quicklook	0.88 (14/480)	goating	1.93 (0/68)

where $f(t)$ is the occurrence of a tag over all intervals, and $f_\Delta(t)$ is its occurrence in interval Δ. C is a regularization constant that increases robustness against scarce observations. In contrast to our own method which estimates the likelihood of a tag occurrence from the observations of the previous month, we find the *interestingness* measure to rely on the entire history of a tag represented by the overall count $f(t)$. This likely makes the *interestingness* measure more stable than our approach, but impedes the detection of reappearing trends, such as Christmas or even *iPhone*[1].

In order to evaluate the *interestingness* measure with respect to our own method, we set $f(t)$ to the sum of current and last month's counts ($f(t) = f_\Delta(t) + f_{\Delta-1}(t)$). Table 9.2 shows the tags with the highest *interestingness* score for November 2007, with C set to 50. We see that the most interesting tags identified by the measure overlap with the top tags detected by our own approach. Moreover, we find the measure to be computationally inexpensive and easy to implement in a real system. However, in contrast to our method, it only performs well if the overall number of tag assignments remains almost constant over consecutive intervals. This however, limits its general applicability to fast growing communities, such as Delicious.

Hotho et al. [HJSS06b] also try to identify trends within social bookmarking services. Instead of using occurrence based measures, they propose to compare the global rankings of tags at different points in time. Rankings are determined using their FolkRank implementation (see Chapter 6). The *popularity change* given the ranking of an object in two consecutive time frames is defined by the authors as

$$pc_{\delta_0 \to \delta_1}(t) = (\frac{r_0(t)}{n_0} - \frac{r_1(t)}{n_1}) \log_{10}(\frac{n_1}{r_1(t)}), \tag{9.6}$$

[1]The term "iPhone" originally referred to a product of Cisco Systems before it was used by Apple (`http://blogs.cisco.com/news/comments/update_on_ciscos_iphone_trademark/`). Both events, the introduction by Cisco and the later adoption by Apple, reappear in the trend curve of the "iphone" tag in our Delicious dataset.

where $r_x(t)$ represents the rank of t at time δ_x, and n_x is the total number of known tags at time δ_x. Since FolkRank delivers rankings for all types of objects, it is equally possible to calculate the *popularity change* for items and users.

For our evaluations we investigate how *popularity change* behaves when tags are ranked simply by the attention they have attracted. We find that in this case *popularity change* strongly favors tags which have been previously unknown, whereas tags above a certain popularity tend to be ignored by the measure independent of their performance. This is likely a result of the power-law distribution of tags. The majority of tags will appear at most once or twice within each month. A small increase in occurrence therefore results in a much better rank for formerly unknown tags, leading to a high increase in the differential term $(\frac{r_0(t)}{n_0} - \frac{r_1(t)}{n_1})$ from Equation 9.6. This increase is not fully compensated by the $\log_{10}(\frac{n_1}{r_1(t)})$ term that favors highly ranked tags. A parameter that would allow to tune absolute versus relative improvements, like the ϵ parameter of our model, is missing. The Top 5 tag trends of November 2007, when ranked by *popularity change*, are presented in Table 9.2.

One important advantage of the FolkRank trend detector over our method is its capability to rank objects with respect to a given context. This enables applications like topic-aware or personalized trend detection, which cannot be realized with our model without modifications. The extension of our model in order to incorporate context is one very promising direction for future research.

9.6 Chapter conclusions

In this chapter, we presented a new probabilistic approach for the emerging problem of tag-based trend detection in folksonomies. Our detector was designed based on sound statistical foundations and considered trends statistical anomalies. We further applied smoothing in order to cope with previously unobserved tags and introduced the smoothing parameter ϵ that lets us control the trade-off between relative and absolute changes in tag diffusion. Our experiments on a large corpus of social bookmarks from Delicious showed that our method successfully detects trends within the tagging behavior of the community. Although we have limited our discussion to the problem of trend detection based on tags, our model can be easily applied to all types of count-based data and could also help to detect user- or item-based trends. We compared our method to previously proposed trend detectors and found that the principled design of our model and the use of priors results in an increased robustness and better scalability to large datasets compared to these methods. A further contribution of this chapter was the application of our attention-based *diffusion-of-attention* concept. Moving from occurrence to attention-based models reduced the potential impact of spam on the quality of detected trends drastically.

Chapter 10

Related work II

This chapter reviews existing work in the field of folksonomy recommendation. It starts with a detailed discussion of previous work in the area of folksonomy recommendation. The following sections present previous work on latent topic models, advanced tag models, and trend detection, since these research directions are closely related to our studies in the Chapters 7 – 9.

10.1 Recommendation in folksonomies

Folksonomy recommendation has become a vibrant research area. Until recently, attention was mainly given to tag recommendation, whereas item recommendation and social search scenarios seemed less popular. However, this picture has changed during the last few years when a lot of publications have started to explore the potential of folksonomies for user interest modeling and personalized content retrieval. On the other hand, scenarios which consider the recommendation of users are still scarce, and the identification of relevant users mainly appears as an intermediate step in item recommendation scenarios.

General approaches

Most works on folksonomy recommendation is restricted to either item recommendation, tag recommendation, or expert finding. However, some authors have approached the task of folksonomy recommendation in general, proposing concepts and methods that can operate in more than one recommendation setting.

The FolkRank algorithm

One of the first publications that investigates the problem of recommendation in folksonomies is **Hotho et al.** [HJSS06a]. The authors consider the general problem of ranking folksonomy objects with respect to a given context. They introduce the Adapted PageRank and FolkRank algorithms which extend the well known PageRank algorithm [BP98] to the tripartite structure of folksonomies. Similar to Pagerank, the authors apply a weight diffusion process that resembles the *random surfer* concept, assuming a user who randomly browses the undirected edges of the folksonomy graph. Folksonomy objects are then weighted by the probability that they will be reached by this *random surfer*. A user is allowed to navigate from items to tags and users, from tags to users and items, and from users to items and tags, which simulates the discovery process commonly found in folksonomies and described in Section 1.3. The diffusion matrix is built from all bipartite subgraphs of G[1]:

$$\mathbf{A}_{FR} = \begin{pmatrix} 0 & \mathbf{A}_{IT} & \mathbf{A}_{IU} \\ \mathbf{A}_{TI} & 0 & \mathbf{A}_{TU} \\ \mathbf{A}_{UI} & \mathbf{A}_{UT} & 0 \end{pmatrix}.$$

The weight update for the **Adapted PageRank** process is the defined as

$$\mathbf{w}_{t+1} \leftarrow d\mathbf{A}'_{FR}\mathbf{w}_t + (1-d)\mathbf{p}.$$

$\mathbf{A}'_{FR}$ is the row normalized, stochastic version of the adjacency matrix $\mathbf{A}_{FR}$, $\mathbf{p}$ is a preference vector that represents context, and d regulates the decay of weight diffusion and thus the impact of the preference vector. Setting all entries of $\mathbf{p}$ to 1 produces a global ranking without context. For context-sensitive rankings the authors suggest to set all entries of $\mathbf{p}$ to 1 except for the entries which correspond to the nodes of the context that are set to $|I|$, $|T|$, or $|U|$, respectively [JMH$^+$07].

Edges in folksonomies are undirected which distinguishes them from links on the web. This circumstance causes direction causes weights to flow forth and back between the same folksonomy nodes, leading to relatively trivial results for the Adapted PageRank algorithm [HJSS06a, JMH$^+$07]. Our own results underline this weak performance, and we find the Adapted PageRank method to perform even worse than simple baseline methods in various recommendation scenarios. To overcome the shortcomings of the Adapted PageRank algorithm Hotho et al. [HJSS06a] propose the FolkRank method.

[1] Transformations of n-partite hypergraphs to a unipartite graph whose edge weights correspond to co-occurrence counts in order to derive object similarities have been previously discussed from a more general perspective by **Xi et al.** [XFF$^+$05].

FolkRank is built on top of the Adapted PageRank algorithm and uses a differential approach

$$\mathbf{w} = \mathbf{w}^1 - \mathbf{w}^0,$$

where $\mathbf{w}^1$ and $\mathbf{w}^0$ are the weights produced by the Adapted PageRank algorithm with and without preferences. Our own evaluations in various recommendation settings show that the FolkRank method produces significantly more accurate recommendations than the Adapted PageRank algorithm. This observation supports the findings of other authors [HJSS06a, JMH+07, RBMNST09, SNM08]

The universality of the FolkRank algorithm makes it a potential candidate for nearly every recommendation setting, and Hotho et al. [HJSS06a] present results for ranking items, tags, and users with respect to a given tag or URL. Additional studies by Jäschke et al. [JMH+07] also report reasonably good results in tag recommendation. However, this universality of FolkRank is traded for accuracy, and our experiments, as well as the evaluations by other authors [RBMNST09, SNM08], show that more specialized recommenders constantly achieve better performances.

One big advantage of the *random surfer* analogy is its simple extensibility beyond tripartite scenarios. For instance, Abel et al. [ABB+09] propose a GFolkRank algorithm that additionally incorporates information about the group a bookmark was posted to. The success story of the FolkRank algorithm further includes its common usage as benchmark method [ABB+09, RBMNST09, SNM08, WSZ09].

The *random surfer* model has been very successful in the web domain. However, its direct transfer to folksonomies is not free of problems. One obvious conceptual shortcoming is the missing directedness of folksonomy edges. Hotho et al. [HJSS06a] note this weakness of their analogy and replace the original Adapted PageRank algorithm by FolkRank. However, the differential approach behind FolkRank appears less principled. This is highlighted when we consider situations without context information. Here, $\mathbf{w}^0$ is equal $\mathbf{w}^1$, such that the FolkRank algorithm will assign zero weights to all objects. An expected ranking in this situation should instead favor popular objects.

Another objection against Folkrank is that the reduction of the folksonomy graph G to a set of bipartite matrices looses information. As a result, we find that recommenders that avoid this reduction, such as our translational model presented in Chapter 8, produce more accurate suggestions. Furthermore, the FolkRank algorithm ignores the user-centricity of tagging. By incorporating the matrix $\mathbf{A}_{IU}$ into the diffusion process, the algorithm implicitly assumes users to be connected to each other if they use the same tags, whereas users with different tagging strategies appear dissimilar. As shown in our own studies, this assumption likely leads to suboptimal recommendations. Another objection against FolkRank is that all inter-object relationships within the matrix $\mathbf{A}_{FR}$ are weighted the same, despite their differences in nature. This assumes that a *random surfer* has no preferences whether to follow items, tags, or users. An assumption that is not proven by empirical evidence. Instead, methods that allow to parameterize the importance of inter-object relationships were shown to achieve improved results in other settings, e.g. in web page ranking [XFF+05].

Hubs and authorities: The SPEAR algorithm

A very interesting approach to the challenge of topic-aware user and item recommendation is presented by **Noll et al.** [NAYG+09]. The authors present the **SPEAR** algorithm which discovers authorities among the users of a folksonomy. SPEAR is built on top of the well-known HITS algorithm that has proven to be a powerful ranking mechanism in various graph-based scenarios [Kle99]. The authority of a user is thought of as topic-dependent, where topics are represented by tags. The authors then consider an expert finding scenario and apply their algorithm to identify the users who possess the highest expertise with respect to a given tag t. For this purpose, they first reduce the folksonomy tensor $\boldsymbol{\mathcal{Y}}_{ITU}$ to the bookmarks that contain the respective tag, resulting in a matrix $\mathbf{A}_{IU}(t) = \boldsymbol{\mathcal{Y}}_{ItU}$. $\mathbf{A}_{iu}(t)$ will be 1 if user u has bookmarked item i with tag t, and 0 otherwise. Further, the authors assume that a user possesses more expertise with respect to a given item if she bookmarks this item before other users. Users who discover items before these become popular within the community are therefore assigned higher weights in $\mathbf{A}_{IU}(t)$. This produces a matrix $\tilde{\mathbf{A}}_{IU}(t)$ with each entry defined as

$$\tilde{\mathbf{A}}_{iu}(t) = score(rank(u, i, \mathbf{A}_{IU}(t))).$$

The matrix $\tilde{\mathbf{A}}_{IU}(t)$ is then the source of an iterative weight diffusion process:

$$\mathbf{u} = \mathbf{i} \times \tilde{\mathbf{A}}'_{iu}(t)$$
$$\mathbf{i} = \mathbf{u} \times \tilde{\mathbf{A}}_{iu}(t).$$

The vectors $\mathbf{u}$ and $\mathbf{i}$ contain the authority and hub weights for users and items. The authority of a user increases if she has been among the early discovers of influential items, i.e. items that act as hubs. On the other hand, the influence of an item increases if it is bookmarked by users of high authority. This mutual reinforcement process is identical to the HITS algorithm by Kleinberg [Kle99].

The authors evaluate the validity of their approach in an expert finding scenario and put a particular focus on the suppression of spam users. Their experiments with artificially injected expert users and spammers show that the SPEAR algorithm will generally recommend expert users first. Spammers will appear at the end of the recommendation list, since they tend to bookmark less popular items. The authors do not evaluate the performance of their method in an item recommendation scenario, though they note the potential of the SPEAR algorithm for item retrieval.

The SPEAR algorithm appears somewhat similar to FolkRank in that both methods reinforce weights among users if they share the same items. However, the SPEAR algorithm reduces the folksonomy graph to only those edges that contain the respective input tag. This excludes all evidence from related tags, loosing potentially important information. For instance, looking for experts on "web development" will not consider edges tagged as "webdev". As a result, users that are experts within a given topic, but use their individual tagging strategy will be excluded by the algorithm. This not only excludes potential expert candidates, but also leads to an unnecessary high sparsity of the $\mathbf{A}_{IU}(t)$ matrix. A technique similar to our translational approach from Chapter 8 could help to integrate relevant tags instead. Tag translation would also allow to involve users with an individual tagging behavior. Furthermore, after the initial graph reduction, the SPEAR algorithm solely considers usage information and disregards further evidence coming from tags. Tags thus cannot act as connectors between items and users. This exclusion of tags misses the chance for more robust and accurate recommendation models, as shown by our own evaluations. Also, the authors do not provide empirical evidence that supports their assumption about the relationship between a user's expertise and the time she bookmarks an item. Users that bookmark items first do not necessarily have to be experts, but could simply be subscribers of a certain type of information channel, e.g. the feed about upcoming items on Delicious.

Ternary PLSA

In an approach that is conceptually similar to our HyPLSA model, **Wu et al.** [WZY06] propose a ternary latent topic model in the spirit of PLSA. The authors assume independence between item, tag, and user observations once the latent concept (intention) behind a tag assignment is know. The probability of a tag assignment is then given as:

$$p(i,t,u) = \sum_{Z} p(z)p(i|z)p(t|z)p(u|z).$$

The plate notation for this probabilistic model is displayed in Figure 10.1. Mappings of latent topics to objects are determined by optimizing the log-likelihood

$$L = \sum_I \sum_T \sum_U \boldsymbol{\mathcal{Y}}_{itu} \log \sum_Z p(z)p(i|z)p(t|z)p(u|z),$$

using the EM algorithm. This produces the estimates $P(i|z)$, $P(t|z)$, and $P(u|z)$ for mapping latent topics to objects. Using Bayes rule these mappings can be transformed to the conditional probabilities $P(z|i)$, $P(z|t)$, and $P(z|u)$. Based on these estimates, the authors then approximate the likelihood of items in a social search scenario:

$$p(i|t) = \sum_Z p(i|z)p(z|t)$$

as well as in a personalized social search scenario:

$$p(i|u,t) \propto \sum_Z p(i|z)p(u|z)p(t|z)p(z).$$

Despite the straightforward extension of the PLSA model to a ternary hypergraph setting, we believe the ternary conditional independence assumption to be unrealistic. Contrary to our own findings, the model implicitly assumes that different users will apply the same tags for the same concepts. Our HyPLSA model avoids this problem by explicitly not including the matrix $\mathbf{A}_{TU}$ into the model building process.

For the social search scenario, the authors do not provide any empirical evidence that the inclusion of users improves the quality over a simple tag-based model. Moreover, the authors do not present an evaluation of their model in a personalized social search scenario, such that the capability of the model in this setting remains unclear. Comparisons to other approaches, e.g. FolkRank, are also missing.

Our HyPLSA method allows for calibrating the influence of collaborative filtering versus tag-based evidence by using the α parameter. Our evaluations show that the setting of this parameter strongly impacts recommendation accuracy. A similar parameter does not exist in the model by Wu et al.. Item, tag, and user observations are thus expected to be of equal quality.

Finally, we note differences in scalability. The ternary latent model is estimated over all tag assignments, leading to a computational effort that grows exponentially by time. Instead, the HyPLSA model works on bipartite aggregations which reduces the required training time.

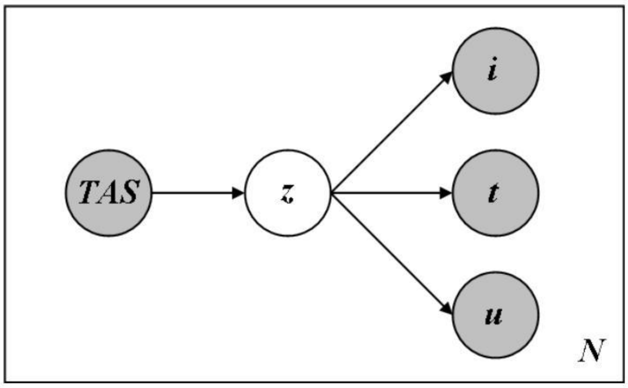


Figure 10.1: Plate notation of the aspect model by Wu et al. [WZY06]. Each tag assignment (TAS) is assumed to be motivated by a mixture of latent aspects from Z. Item, tag, and user occurrences are assumed independent with respect to Z.

Tensor decomposition

Instead of reducing the folksonomy hypergraph to bipartite co-occurrence graphs, **Symeonidis et al.** [SNM08] propose to decompose the full folksonomy tensor and estimate missing values using Higher Order SVD (HOSVD). HOSVD generalizes SVD[2] to higher dimensional tensors.

SVD factorizes a matrix $\mathbf{A}$ as a product $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}'$, where $\mathbf{\Sigma}$ is a diagonal matrix that contains the singular values of $\mathbf{A}$, and $\mathbf{U}$ and $\mathbf{V}$ are input-output transformation matrices. $\mathbf{A}$ can be approximated as $\tilde{\mathbf{A}} = \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}'$, where $\tilde{\mathbf{\Sigma}}$ is the same as $\mathbf{\Sigma}$ except that values which do not fall among the K highest values of $\mathbf{\Sigma}$ are set to 0. This approximation minimizes the Frobenius norm $||\mathbf{A} - \tilde{\mathbf{A}}||_F$ for a given value of K.

Similar to standard SVD, HOSVD can be applied to factorize the folksonomy tensor $\boldsymbol{\mathcal{Y}}_{ITU}$ as

$$\hat{\boldsymbol{\mathcal{Y}}}_{ITU} = \boldsymbol{\mathcal{C}}_{I'T'U'} \times_{I'} \mathbf{A}_{I'I} \times_{T'} \mathbf{A}_{T'T} \times_{U'} \mathbf{A}_{U'U},$$

where $\boldsymbol{\mathcal{C}}_{I'T'U'} \in \mathbb{R}^{|I'|\times|T'|\times|U'|}$ is a (reduced) core tensor and $\mathbf{A}_{I'I}$, $\mathbf{A}_{T'T}$, and $\mathbf{A}_{U'U}$ are transformation matrices for items, tags, and users respectively[3]. Analogous to SVD, HOSVD tries to minimize the element-wise quadratic

[2]see Chapter 7

[3]For details on HOSVD and tensor decomposition in general refer to [DLDMV00] and [KS08]. For tensor decomposition in a folksonomy context see [SNM08] and [RBMNST09].

difference between the original and the approximated tensor:

$$\operatorname{argmin} \sum_{I} \sum_{T} \sum_{U} (\boldsymbol{\mathcal{Y}}_{itu} - \hat{\boldsymbol{\mathcal{Y}}}_{itu})^2.$$

After the decomposition each value of $\hat{\boldsymbol{\mathcal{Y}}}_{ITU}$ can be calculated as

$$\hat{\boldsymbol{\mathcal{Y}}}_{itu} = \sum_{I'} \sum_{T'} \sum_{U'} \boldsymbol{\mathcal{C}}_{I'T'U'} \mathbf{A}_{i'i} \mathbf{A}_{t't} \mathbf{A}_{u'u}.$$

The approximation tensor $\hat{\boldsymbol{\mathcal{Y}}}_{itu}$ will mainly contain non-zero entries which permits the ranking of all type of objects. For instance, in a personalized search scenario items can be recommended based on their value in the vector $\hat{\boldsymbol{\mathcal{Y}}}_{Itu}$. Symeonidis et al. instead consider the task of tag recommendation and weight each tag for a given bookmark $b = (i, u)$ based on its value in $\hat{\boldsymbol{\mathcal{Y}}}_{iTu}$. They report improved accuracy of their method compared to a FolkRank recommender.

However, we were unable to replicate these positive results in our own experiments. Instead, we found HOSVD to produce much less accurate recommendations than most other methods, including FolkRank, independent of the recommendation scenario. Other authors confirm these findings. Rendle et al. [RBMNST09] report that the high sparsity of the original $\boldsymbol{\mathcal{Y}}_{ITU}$ tensor is a severe obstacle for this type of tensor decomposition. They argue that the strong dominance of zero-entries in $\boldsymbol{\mathcal{Y}}_{ITU}$ will lead to a considerable bias towards unobserved data. However, these shortcomings of HOSVD do not question the general value of tensor decomposition for folksonomy recommendation. Instead, Rendle et al. [RBMNST09] propose the use of ranking-based optimization criteria during the tensor decomposition process, replacing common least-square criteria, such as HOSVD. As a result, the authors present an approach to tag recommendation, where the quality of the decomposed tensor is estimated in terms of AUC [RBMNST09]:

$$\operatorname{argmax} \sum_{(i,u) \in B} AUC(i, u, \hat{\boldsymbol{\mathcal{Y}}}_{ITU}).$$

They report a slightly improved performance of their tag recommendation model compared to FolkRank. However, this optimization will generate less universal recommendation models than HOSVD, since different recommendation scenarios will require other optimization criteria.

The potential of ranking-based tensor factorization models in the context of folksonomy recommendation is emphasized by the fact that Rendle et al. won the 2009 ECML PKDD Discovery Challenge on tag recommendation

[RST09]. Nevertheless, the high computational complexity required for the training of tensor-based recommenders remains a severe obstacle for their application to real-world folksonomies [RBMNST09].

Item recommendation

Various authors present solutions to the specific problem of item recommendation in folksonomies. **Bao et al.** [BXW+07] assume a mutual enhancement relation among popular items, up-to-date web users, and 'hot' social annotations. Similar to the FolkRank algorithm, they propose a weight diffusion process along the bipartite subgraphs of a folksonomy in order to estimate rankings for websites. Their ranking method (*SocialPageRank*) calculates item rankings based on a diffusion matrix $\mathbf{D}$. This matrix is constructed as product of all bipartite folksonomy matrices:

$$\mathbf{D}_{II} = \mathbf{A}_{IU}\mathbf{A}_{UT}\mathbf{A}_{TI}\mathbf{A}_{IT}\mathbf{A}_{TU}\mathbf{A}_{UI}.$$

Weights thus flow from items to users to tags to items and then the same way back to items. This distinguishes the setup from FolkRank, where weights always flow from one type of folksonomy object to the other two types. Item weights are then iteratively calculated as $\mathbf{i} = \mathbf{D}_{II} * \mathbf{i}$, until the convergence of $\mathbf{i}$. User studies showed the *SocialPageRank* to produce more accurate website rankings than the PageRank from Google. The authors use an additional diffusion process on the $\mathbf{A}_{IT}$ matrix to obtain tag-tag and item-item similarities (*SocialSimRank*). These similarities are then utilized for query reforming, but no individual tag vocabularies are considered. A comparison to other folksonomy algorithms, such as FolkRank, is missing. Furthermore, the selection of the diffusion process beyond the *SocialPageRank* seems heuristic and is not compared to other potential ways of diffusion. The inclusion of the $\mathbf{A}_{TU}$ matrix in this process is problematic due to the heterogeneous tagging behavior of users.

Abel et al. [ABB+09] propose a generic contextualization method for ranking objects in folksonomies and consistently model context in the form of tag clouds. Based on these context clouds, they can re-rank all types of recommended objects within their GroupMe! folksonomy[4] independent of the original ranking algorithm. The authors consider three sources of context: the actual user u modeled by T_u, the resource i a user has visited last represented by T_i as well as the GroupMe! group resource i belongs to, represented by the most frequent tags in this group. The authors then consider a personalized social search scenario, and evaluate which kind of

[4]http://groupme.org

context provides the most valuable information about a users next steps. Further, they compare the performance of different ranking algorithms, such as Folkrank, which had to predict relevant items based on the given context data and a query tag. The most interesting observation of their evaluations is that the last visited resource provides much better information about a user's next activities than her personomy tags. An exploitation of this observation has the potential to further improve the recommendation models presented in this dissertation.

The consistent modeling of folksonomy objects as tag clouds simplifies the recommender design. However, the authors do not study whether tag clouds of different object types behave similarly. For instance, our own evaluations show a major difference between the concentration of tags in personomies and item vocabularies (see Section 2.3). Direct comparisons of these vocabularies thus likely lead to suboptimal results. Modeling users not by their own tag cloud, but by the translation of this cloud may solve this problem.

Zanardi and Capra [ZC08] also approach the problem of content recommendation in folksonomies. Ignoring inter-user differences in tagging behavior, they construct a collaborative filtering model, where the similarity between users is determined by the cosine between their personomies. Furthermore, they calculate the similarity between tags based on their cosine similarity in the item space. Combining tag and user similarities into a unified model, the authors then expand tag-based queries and personalize recommended CiteULike publications. A comparison to other recommendation algorithms is missing.

The item recommender by **de Gemmis et al.** [dGLSB08] combines tags and document terms to derive user interest models. In contrast to most other approaches, the authors do not only consider positive tag/term samples, i.e. the tags/terms from the items a user liked, but also introduce a notion of negative tags/terms, i.e. common tags/terms for items the user did not like. Like and dislike are explicitly specified by the users by rating items during the tagging process. The authors then train one Naïve Bayes classifier based on the terms/tags of positive samples and a separate one for negative samples. These classifiers are used to personalize search results with respect to the user. Modeling users not by the tags of their own personomy, but by the tag clouds of the bookmarked items is very similar to the result one would get when translating personomies with our translational model from Chapter 8. However, the authors do not investigate this aspect of their approach, nor do they compare their method to other recommender models.

A hierarchical clustering technique that groups tags based on their similarity in the item space is presented by **Shepitsen et al.** [SGMB08]. Items and users are folded into the cluster space based on the overlap between the item and personomy tags with the tags of each cluster. Overlap is defined at a set level as the ratio of tags from a vocabulary that also appear in a cluster. User-level differences in tag usage or the problem of homonyms are not considered during the model building process. Clusters are then used to improve the accuracy in a personalized social search scenario. Item weights are calculated as the product of an item's similarity to the query tag(s) and its similarity to the user within the cluster space. Evaluations showed positive effects of their clustering method on a dataset from the Last.fm music streaming folksonomy and less satisfying results for a Delicious sample of higher sparsity. Both datasets used in the publication are proprietary, and a comparison to existing methods is missing.

An identical problem setting and a very similar approach is presented by **Xu et al.** [XBF+08]. Here, the authors determine the ranking of an item from a mixture of its rank with respect to the query tag, using a *tfidf*-based distance measure [MRS09], and its rank with respect to the user, which is determined within a topic space. The authors compare two ways of topic creation. The first version simply considers item-related tags as topics, whereas the second uses the category labels obtained from the website hierarchy of the Open Directory Project. The authors then apply a weight diffusion technique which smoothens topics between users and items if these are connected within the matrix $\mathbf{A}_{IU}$. Their results state that annotation-based topic models produce more accurate recommendations than category-based models. Diffusing the tags among the users with similar bookmarking behavior is likely to generate more robust user tag models that are less vulnerable to user-level differences in tagging behavior. This makes the approach similar to our translational model. However, our own approach is more principled. A comparison to benchmark methods or evaluations on publicly available folksonomy datasets are missing.

Sen et al. [SVR09] introduce a tag-based movie recommender. They investigate the potential of tags in two common scenarios: rating prediction, where the algorithm has to predict how a user will rate a movie, and preference prediction, where the algorithm has to rank movies with respect to a user's preferences. The authors assume that a user will rate/prefer movies the same way if these exhibit similar tag spectra. They design a selection of tag-based recommendation models and compare their performance to *state-of-the-art* collaborative filtering algorithms. Their results show an improved accuracy for tag-based models compared to collaborative filtering for the preference ranking task, whereas collaborative filtering methods produced more accurate rating predictions. The best predictions in both scenarios were given by a hybrid model that combined tags with collaborative filtering.

The integration of tags into movie recommenders is also discussed by

Szomszor et al. [SCA+07]. The authors query external services, e.g. 'The Internet Movie Database' (IMDB)[5], and construct tag clouds for all films a user rated equally. A rating prediction for a previously unrated movie is then made based on the similarity of its tags to a user's rating tag clouds. The authors report that their annotation-based method performed better than simple baseline predictors, but a comparison with more advance collaborative filtering methods is missing. However, the integration of tags in movie recommendation may prove especially useful in cold-start situations, where a movie has not yet received a sufficient number of ratings.

Siersdorfer and Sizov [SS09] utilize Latent Dirichlet Allocation (LDA) to cluster pictures of Flickr based on tags. Users are then folded into the clusters via their personomy tags. This allows the authors to describe items, tags, and users in a unified latent space, which in turn enables them to derive similarities between different types of objects, e.g. items and users. Their approach is thus somewhat similar to our HyPLSA model. The authors apply their latent model in order to produce personalized item (photo) recommendations as well as for user and group recommendations. Clustering users based on their personomy tags alone ignores individual tagging strategies and will therefore lead to less optimal results. Furthermore, since Flickr is a narrow folksonomy, item vocabularies will suffer from the same heterogeneity as personomies, such that a clustering of items based on tags is problematic. Nevertheless, Latent Dirichlet Allocation was shown to outperform PLSA on various tasks, see e.g. [BNJ03]. An extension of our PLSA approach to LDA could thus be a promising direction for future research.

Li et al. [LHY+09] try to solve the problem of tag ranking for Flickr. They argue that the order of user-assigned tags not necessarily reflects the relevance of tags from a retrieval perspective. The authors therefore propose to re-rank tags by their relevance with respect to the labeled content. They estimate this relevance using a weight diffusion technique over the tag-tag graph. This graph is constructed based on two similarity measures. The first tag-tag similarity is calculated by comparing visual features of the photos that link to a tag, whereas the second similarity is derived by comparing tags in the item space. Tags of a photo can then be ranked based on which other tags have been assigned to it. The authors note that the identification of relevant content descriptors is crucial for scenarios such as tag-based image search. Even though the problem of tag re-ranking mainly applies for narrow folksonomies such as Flickr, where content is labeled by a single user, broad folksonomies face similar problems for items that have not yet been bookmarked by many users. Narrow folksonomies especially suffer from

[5] http://www.imdb.com

individual tagging habits, since these are not compensated by cumulative effects. This makes the design of search solutions for these environments an even more challenging task. Similar to our translational model, correlating tags with content features in order to infer the meaning of individual tags and translate them to a global representation could be a first step in the right direction.

Tag recommendation

One of the first works on graph-based tag recommendation in folksonomies is **Jäschke et al.** [JMH+07]. The authors compare the performance of co-occurrence-based tag recommenders and more complex recommenders based on the FolkRank algorithm [HJSS06a]. For FolkRank, they set the preference vector $\mathbf{p}$ to $\mathbf{1}$ except for the user and item the tags have to be recommended for, and which are set to $|U|$ and $|I|$ respectively[6]. An evaluation of other possible settings for $\mathbf{p}$ is not provided. A parameter that controls the influence of an item with respect to a user, similar to the α and β parameters in our experiments in Chapter 8, is also missing. The authors report only minor improvements of the FolkRank method over baseline co-occurrence approaches. Our own experiments on tag recommendation show the FolkRank recommender to perform even worse than these baselines (see Chapter 8).

Most similar to our work on translational tag models from Chapter 8 is **Lipczak** [Lip08]. The author proposes a tag recommender that combines the actual tags previously assigned to a resource with the tags from a user's personomy as well as with keywords extracted from the title of a Bibsonomy publication or URL. Similarly to our translational model, user tags are chosen based on the tags that have been previously assigned to an item by the community. The mapping between personomy and community tags is achieved using a lexicon-based approach, but no further analysis of this approach is presented.

Other strongly related work comes from **Gemmell et al.** [GSRM09, GSRLC09] who propose a K-Nearest Neighbor algorithm [MRS09] for tag recommendation in folksonomies. One of their methods recommends tags for an item i by looking at how the same user tagged similar items in the past. This is algorithmically very related to our translational approach, since we recommend tags from a user's personomy if these had a previous high co-occurrence with the tags of i. The item-centric collaborative filtering approach achieved a recommendation quality similar to the one of a FolkRank algorithm in evaluations on Delicious and CiteULike. **Guan et. al** [GBM+09] construct their

[6] This is the same configuration of the FolkRank algorithm as in Chapter 8

tag recommender in a similar way. They start by calculating item-item similarities in the user space. Personomy tags are then weighted higher if the user had assigned them to similar items previously. Though the solutions from [GBM+09] and [GSRM09, GSRLC09] will produce recommendations that are similar to the ones of our tag translation model from Chapter 8, we believe that our translational model is more principled and universally applicable beyond tag recommendation alone. Instead it can be incorporated into other types of tag-centric scenarios as we have exemplary shown for tag-based social search in Chapter 8.

Instead of reducing the folksonomy hypergraph to bipartite co-occurrence graphs, **Symeonidis et al.** [SNM08] propose to decompose the full folksonomy tensor and estimate missing values using Higher Order SVD (HOSVD). They claim reasonably good results on the task of tag recommendation. **Rendle et al.** [RBMNST09] report an even better performance when using ranking-oriented tensor factorization methods. A similar tensor factorization model by the same authors won the 2009 ECML PKDD Discovery Challenge[7] [RST09].

Graph-based approaches to tag recommendation fail in cases of new items or users, which arguably represent a vast amount of all objects. Many solutions try to overcome these local cold-start problems by incorporating content information into the model building process. For instance, **Song et al.** [SZL+08] construct matrices for item-tags ($\mathbf{A}_{IT}$) and item-words ($\mathbf{A}_{IW}$) co-occurrence observations. Similar to the FolkRank graph construction, they then fuse both matrices into one matrix of the form

$$\mathbf{A}_{Song} = \begin{pmatrix} 0 & \mathbf{A}_{IT} & \mathbf{A}_{IW} \\ \mathbf{A}_{TI} & 0 & 0 \\ \mathbf{A}_{WI} & 0 & 0 \end{pmatrix}.$$

This matrix is factorized using a low rank approximation technique. Tag recommendations for new items are calculated by folding items into the reduced space via their term distributions. Personalization aspects are not considered.

Heymann et al. [HRGM08] investigate to what degree different aspects of web pages, such as the page text, anchor texts, or surrounding hosts, e.g. the domain of a page, can help improve the quality of a Delicious tag recommender. They report that many labels can also be predicted from content features which may especially help to identify valid tags in the 30 to 50 percent of cases [HRGM08] where URLs had been posted only once or twice. The authors further use association rules to harvest relations between

[7] http://www.kde.cs.uni-kassel.de/ws/dc09/

tags. These association rules are then utilized to augment the recall of single tag queries. The association rules between tags serve a similar purpose as our translational approach. However [HRGM08] discuss global mappings only, whereas our approach focuses on user-level mappings.

Further work on hybrid tag recommenders which incorporate annotations and content or meta-data in parallel is presented by **Tatu et al.** [TSD08] and **Lipczak et al.** [LYHM09], the winners of the ECML PKDD Discovery Challenge in 2008 and 2009. Both teams build tag recommenders for Bibsonomy, combining information from an item's meta-data, such as the title of a publication, its authors, or the abstract, with graph-based evidence, i.e. a user's previous tagging behavior. They report an improved recommendation quality for their hybrid models. This quality was confirmed by the strong performance during the challenges.

An extreme way of incorporating external sources for better recommendations is described by **Mrosek et al.** [MBA+09]. Here, the authors consider the problem of tag recommendation for new items, but ignore all additionally given meta-data. Instead, they argue that most items have already been tagged in other tagging communities. Their solution retrieves tags from external services and merges all sources into a single recommendation set. This approach performed surprisingly well and came in second in the 2009 ECML PKDD Discovery Challenge.

Tag recommendation not from a quality but implementation-oriented perspective is discussed by **Jäschke et al.** [JEHS09]. The creators of the Bibsonomy bookmarking service describe the practical challenges when integrating tag recommendations into a real tagging system. They also report that only one third of all tested Bibsonomy users actually clicked on the tags that had been recommended. This low percentage was mainly caused by new users without many postings, whereas frequent users showed more interest in the provided recommendations.

There also exists a body of work about tag recommendation in narrow folksonomies, mainly on Flickr. **Wu et al.** [WYYH09] analyze how valid tags can be derived from visual content features extracted from Flickr pictures. **Garg and Weber** [GW08] present an interactive tag recommender for Flickr that adapts recommendations whenever the user enters a new tag. They then discuss the applicability of common IR techniques in such a scenario. Finally, **Weinberger et al.** [WSVZ08] investigate the problem of tag disambiguation among Flickr tags. They identify situations of ambiguous tag usage and recommend disambiguating tags to the user for clarification. They verify the performance of their approach on geographical, temporal, and descriptive tags.

All tagging services discussed throughout this work already offer some type of tag recommendation. These suggestions influence the way people tag. However, previous work on tagging and especially tag recommendation considers tags to be created by mental processes alone, disregarding the impact of existing recommendation functionality. Consequently, it has become common practice to evaluate novel tag recommenders not during the tagging process itself but on posterior data. An optimal recommender in such a setting does not necessarily need to predict the tags a user would choose herself. Instead, it has to predict the tags a user assigns when a (generally unknown) list of recommendations is presented to her. So far, research has not paid much attention to the actual impact of existing tag suggestions, with one exception being the work of **Suchanek et al.** [SVG08]. The authors present a large-scale user study and report that around one third of all tags is actually selected from a list of recommended tags. They further note that tags in the beginning of the recommended list have a higher probability of being selected, even though differences are marginal. More realistic evaluations of tag recommendation approaches may come from on-line scenarios, where recommendations are directly evaluated by user feedback, as set up during the 2009 ECML PKDD Discovery Challenge.

User recommendation

User recommendation in folksonomy environments has still not attracted as much research attention as tag or item recommendation. Instead, most authors utilize the identification of relevant users merely as intermediate step in other scenarios.

The previously discussed approach by **Noll et al.** [NAYG+09] is one example of how the interaction between users and items in a folksonomy can be exploited for the expert finding task. The authors report that their method is able to successfully distinguish between artificially created expert users, followers, and newcomers. A useful side effect of their approach is the suppression of spam activity, since spammers are unlikely to identify relevant content before experts. **Abel et al.** [ABB+09] also investigate the applicability of their method to the task of user recommendation. The authors design different ranking algorithms. One of these algorithms, called *SocialHits*, is similar to the SPEAR algorithm by [NAYG+09], in that it derives the authority of a user from the time she bookmarked an item, assuming that domain experts will react to new domain items first. Both algorithms, SocialHits and SPEAR, extend previous work by **Kleinberg** [Kle99] on the detection of *authoritative sources in a hyperlinked environment*. A further common characteristic of both approaches is the way user authority is modeled. A user is considered an authority if many other users have annotated resources this user had annotated before. No further studies on the validity of this assumption are provided.

Siersdorfer and Sizov [SS09] utilize their LDA cluster algorithm to derive interesting users and groups for Flickr users. Users are mapped into the latent topic space based on their personomy tags. The authors then calculate (user) similarities within the latent space. The applicability of LDA to the problem of user interest modeling was also shown by **Zhou et al.** [ZBZ+08].

A topic that is strongly related to user recommendation is the problem of community finding. **Li et al.** [LGZ08] group folksonomy users based on their item collections. Groups are categorized by tag sets. Community detection in the blogosphere is considered by **Java et al.** [JJF08]. The authors fuse linkage and tagging patterns into a unified graph representation and apply graph cutting to find partitions within this new graph. Each partition then constitutes a community of blogs and most relevant community tags. For additional work on community detection in blogs see also [ALTY08].

Finally, **Szomszor et al.** [SCA08] move away from the single-service perspective and aim to match users across multiple tagging communities based on shared tagging patterns. Their evaluation shows that "prominent user interests, important locations, and events, are often reflected in the intersection between tag-clouds, irrespective of the focus of the folksonomy".

10.2 Latent topic models

In Chapter 7 we propose a hybrid probabilistic latent topic model for folksonomies. Probabilistic clustering methods have been previously investigated in many other domains. Probabilistic latent semantic analysis (PLSA) was introduced by **Hofmann** [Hof99b] in 1999, and has since then been successfully applied to diverse tasks, e.g. collaborative filtering based recommendation [Hof99b], document retrieval [Hof99a], automated image annotation [MGP03], document summarization [Hen09], the discovery of navigational patterns on the Web [JZM04], music recommendation [AGMP+07], and news personalization [DDGR07].

Similar to our HyPLSA approach, **Chi and Hofmann** [CH00] consider the problem of document clustering and extend the PLSA algorithm in order to combine content-based and hyperlink-based similarities into a unified model. They show that the accuracy of their hybrid model is greater than the one of either model in isolation. **Popescul et al.** [PUPL01] consider another tripartite scenario and build an aspect model based on the interactions of users and documents as well as the terms contained in each document.

Their scenario is thus somewhat similar to a folksonomy setting, except that words are bound to items, whereas tags are bound to a user-item combination, i.e. a bookmark[8]. The authors then perform Tempered EM in order to fit their model, and report severe overfitting problems due to data sparsity. **Brants et al.** [BCT02] find that overfitting problems can be reduced by training various PLSA models with different random initializations and averaging estimates over all models. Our own evaluations in Chapter 7 support these findings, and averaged models performed consistently better than single models.

A further recently very popular probabilistic modeling approach is the Latent Dirichlet Allocation (LDA) by **Blei et al.** [BNJ03]. The LDA model is very similar to PLSA, but allows for non-uniform Dirichlet priors [BNJ03]. LDA has been successfully applied to many traditional information retrieval problems, such as document modeling and collaborative filtering [BNJ03]. Latent Dirichlet Allocation also becomes increasingly popular for clustering in folksonomies [RHMGM09, SS09, KF09].

10.3 Advanced tag models

Building effective recommender systems on tagging data requires detailed knowledge of how tagging actually works. Consequently, much effort was spent in order to achieve a better understanding of the intentions and dynamics behind tagging, enabling the development of advanced tag models. This dissertation contributes to the work on tag models by proposing a novel approach to tag translation in Chapter 8. It also contributes the first exhaustive study on inter-user differences in tagging strategies.

Cattuto et al. [CBHS08] analyze different ways of measuring tag relatedness in a folksonomy environment. They investigate co-occurrence based metrics, where tags become related if they belong to the same bookmark, as well as distributional metrics that consider tags to be related if their distributions in the user or item space appear similar. Additionally, they apply the FolkRank algorithm to rank tags with respect to another tag in order to derive relatedness. The authors report very different results depending on the selected relatedness measure. They find that distributional methods produce tags which are semantically similar or even synonyms, while co-occurrence and FolkRank deliver rather conceptual related tags. The authors therefore propose to utilize different forms of tag relatedness in tasks such as synonym discovery, concept hierarchy extraction, tag recommendation, and query ex-

[8]This structural difference disappears in narrow folksonomies, where each item belongs to exactly one user.

pansion. One remarkable result, which is directly related to our own work, is the observation that item and user dimensions produce different forms of relatedness. While tags that are assigned to the same items tend to be synonyms or different spelling forms of the same concept, e.g. "web2.0" versus "web20", they find that tags which frequently overlap in personomies are rather conceptionally related, e.g. "web2.0" and "ajax". This further supports the assumption that users will try to avoid using redundant tags, but rather develop standardized tag vocabularies for more efficient content categorization (see Chapter 8).

Very similar is the work by **Yeung et al.** [AYGS09] who investigate the problem of tag contextualization. The authors start by manually defining sets of disambiguating tags that help to identify whether for instance the tag "sf" refers to "San Francisco" or "Science Fiction". They then evaluate how different sources of tag similarity help to disambiguate these manually defined tag pairs. All similarities are calculated after clustering the respective co-occurrence data. They find that clustering compares favorably to external sources for contextualization, such as WordNet. One of the evaluated measures considers two resources as similar if these have been labeled by a user with the same tag. Yeung et al. report that this user-level similarity measure was better adapted for tag disambiguation than other co-occurrence measures. This finding supports the motivations behind our user-centric approach to tag modeling.

Oldenburg et al. look at a broader picture and investigate how tag distributions differ among various folksonomies, such as Delicious, Bibsonomy, CiteULike, and Connotea[9]. Although the authors only compare global distributional patterns in their work, it is obvious that the combination of services can lead to more robust tag models, especially for tags of low frequency within some services. Furthermore, using tag-bridges across different services may allow for the cross-domain correlation of items and users with positive effects on all types of recommendation scenarios.

A further way to enhance tag models is the incorporation of external knowledge. The most prominent external source is the WordNet corpus that helps to identify tag types as well as tag hierarchies. For examples see e.g. [LEC07, OSvZ09, SNCA+09, AYGS09, CBHS08]. Finally, important analytical work leading to a better understanding of the intentions, patterns, and dynamics of tagging comes from **Golder et al.** [GH06], **Bischoff et al.** [BFNP08], **Marlow et al.** [MNBD06], **Halpin et al.** [HRS07], and **Ames et al.** [AN07]. See Chapter 5 for a discussion.

None of the above publications explicitly studies user-level differences in

[9] http://www.connotea.org

tagging – though many authors note their existence.

10.4 Trend detection

Folksonomies also provide a promising source for the detection of trends, even though literature on this topic is still sparse. The work most closely related to our own is **Hotho et al.** [HJSS06b]. The authors apply their FolkRank algorithm to derive rankings for items and tags at different points in time. They then determine the *popularity change* of an object by comparing the rankings from two consecutive intervals. Their algorithm also allows them to detect trends with respect to a given topic (tag). The proposed method is evaluated on a data set of 7.7 million Delicious posts.

Dubinko et al. [DKM+06] visualize tag trends and corresponding photos for Flickr. They propose a straightforward *interestingness* measure defined as $int(i,t) = f_t(i)/(C + f(i))$, where $f(i)$ is the overall occurrence of an item, $f_t(i)$ its occurrence in interval t and C is a regularization constant that increases robustness against scarce observations. Their user interface then adapts the font size of each tag to the calculated *interestingness* factor, such that users can easily identify upcoming topics. The definitions of *interestingness* as well as *popularity change* look rather heuristic, and especially the *interestingness* measure appears trivial. The trend detector we presented in this work is instead built on strong statistical foundations.

Related work on trend detection also comes from other domains. Very similar in nature to [HJSS06b] is the trend detector by **Berberich et al.** [BBVW06] who compare the ranks of publications over time. Ranks are determined by a PageRank-like method. However, their model does not provide topic-aware trend detection functionality as in [HJSS06b]. **Glance et al.** [GHT04] use count-based measures to detect trends within the blogosphere. A more advanced technique for blog trend detection is proposed by **Chi et al.** [CTT06] who apply SVD and HOSVD and detect trends by looking at the dynamics of the eigenvalues of the factorized matrices. **Amitay et al.** [ACH+04] aim to discover trends on the web by performing a temporal link analysis. Their model allows them to detect and track significant trends in the web community.

Algorithmically most similar to our probabilistic approach to trend detection is work by **Kleinberg** [Kle06]. The author discusses a principled probabilistic model for trend detection in information streams. Trends are considered as unlikely events which is also the basic assumption of our approach from Chapter 9. However, our approach goes beyond the discussed method by introducing a smoothing function, and the smoothing parameter

ϵ lets us control the trade-off between relative and absolute changes.

Conclusions

Conclusions

This dissertation aimed to develop novel recommendation solutions for broad folksonomies. This goal was motivated by two major observations:

1. Folksonomies have become a rich source of information.

2. The semantic contribution of tags can help in the design of enhanced recommender systems.

An exhaustive analysis of various real-world folksonomies demonstrated the validity of these assumptions and highlighted the importance of tags as connecting elements between users and content. The second part of this dissertation then presented novel recommendation solutions that were specifically tailored to the needs and the information structure of folksonomies and incorporated the additional evidence of tags. Extensive evaluations of the presented recommender algorithms emphasized the positive effect of tags on the descriptiveness and robustness of the developed models. Our experimental results further supported the intuition that recommender systems become more accurate the more they respect the hypergraph nature of folksonomies. Experiments on several datasets and various recommendation tasks, such as personalized item recommendation, tag recommendation, social search, and tag-based trend detection, showed that the developed methods produce more accurate suggestions than the current *state-of-the-art*.

Summary of contributions

This dissertation has advanced the *state-of-the-art* in folksonomy analysis and graph-based recommendation in collaborative tagging environments through a number of contributions. **Part I** of this dissertation looked at the nature of folksonomies and significantly contributed to a better understanding of the motivations and dynamics behind collaborative tagging as well as their potential impact on folksonomy recommendation.

Chapter 1 started with an introduction of tagging and distinguished tagging from existing categorization techniques. The chapter further described different forms of folksonomies and highlighted the potential of folksonomies for information retrieval.

Chapter 2 presented an exhaustive analysis of the bookmarking and tagging patterns within Delicious based on one of the largest snapshots of a real-world folksonomy ever analyzed in academia. The results presented in this chapter have been published in [WZB08], [WZB10a], and [WZB10b].

Chapter 3 looked at folksonomies from a network perspective and, among other results, found that tags act as highly connected hubs between users and items.

Chapter 4 described the problem of folksonomy spam and characterized the motivations and behavior of spam users. It then presented several novel methods for spam detection and spam suppression. The chapter summarized the contributions of [WZB08], [WZB10a], and [NWO09].

Based on the analytical findings of the first part, **Part II** of this work introduced novel recommendation algorithms that had been especially tailored to the needs of folksonomies.

Chapter 6 presented the first comprehensive academic analysis of potential recommendation scenarios in folksonomies. The chapter introduced a classification scheme for folksonomy recommendation based on different notions of context and applied this scheme in order to categorize and distinguish previously presented approaches to folksonomy recommendation.

Chapter 7 proposed a novel probabilistic model that clustered users, items, and tags of a folksonomy into a common latent topic space. The HyPLSA recommender was trained from usage and tagging information in parallel in order to better cope with the ternary structure of folksonomies. Evaluations on various datasets showed that this hybrid recommender produced better item recommendations than non-hybrid versions and the current *state-of-the-art*. The results underline the potential of tags for folksonomy recommendation. The HyPLSA approach was previously presented in [WUS09]. The analysis of HyPLSA in cold-start environments has appeared in [SWUH09].

Chapter 8 investigated user-level tagging patterns and found that users develop individual tagging vocabularies over time. It then presented an innovative approach to tag translation which mapped the individual tags of a user to a common global language. The approach was evaluated in tag recommendation and tag-based social search scenarios, showing improvements over *state-of-the-art* methods. An initial version of our approach to tag translational was discussed and evaluated in [WSZ09]. However, the chapter mainly reproduced the more detailed analyses and evaluations presented in [WZB10b].

Chapter 9 considered the task of tag-based trend detection in folksonomies and introduced a novel probabilistic approach that considered trends as statistical anomalies. Experiments showed that this trend detector could successfully detect new as well as seasonal trends. This chapter summarized the contributions of [WPK+08] and [WZB10a].

Future research

There are a variety of potential directions for future research in the area of folksonomy-based information retrieval.

Interaction of tagging and retrieval

One area of future research is to examine to what degree the retrieval interface influences the tag assignment process. For instance, a retrieval interface that indexes item features, such as the title of a publication, may reduce the need for assigning tags related to those features. One extreme example of this are solutions, such as the publication managing tools Mendeley[10] and Zotero[11], that offer full-text content search in addition to tagging functionality. Users of these tools will be less motivated to tag content attributes if these have been already indexed by the search engine and can be accessed via the search interface. Consequently, users of services that support full-text searches are expected to develop different tagging habits than users of classical folksonomies, and will likely assign rather qualitative than descriptive tags. Understanding the relationship between tagging and retrieval will not only help in the design of better tag models, but will also allow for the development of adequate retrieval interfaces.

Moreover, a user will assign the tags to an item which she considers relevant at the moment of tagging. However, relevance is not static, and changing tagging strategies, evolving user interests or alternating retrieval tasks may require a constant re-labeling of previous content. This increases the maintenance effort of tagging, compared to other categorization techniques, such as full-text indexing. Reducing those maintenance costs by combining tagging with other retrieval paradigms will be a key challenge for next generation content organization solutions.

[10] http://www.mendeley.com
[11] http://www.zotero.org

Extending the folksonomy graph I

Modeling folksonomies as networks of items, tags, and users is a strong simplification. Instead, real-world folksonomy services produce information that goes well beyond the borders of this model. Potential sources of information for next generation folksonomy recommenders could include:

Additional data about items and users. Users may additionally provide descriptions or meta-data about items. Some services allow users to add bibliographic data, locations, timestamps, textual descriptions, comments etc.. Users may also provide profile data or explicitly state their interests and preferences. All this additional data can help in the design of better item and user models, which in turn, will allow for more accurate recommendations.

Social interactions. Most folksonomies will support social interaction in one way or another. Friendship or subscription networks as well as group functionality may help to derive different types of user similarity based on the social graph and to identify influential users. Recommendations could then be given based on the trends in a user's social environment. For instance, the interests of a user and the interests of her friends in Flickr were shown to correlate [CMG09].

Multi-modal feedback channels. In this work we assumed that users express their interest in an item by bookmarking it. However, feedback can also arrive in different forms, e.g. explicitly, in the form of ratings and comments, or implicitly, when a user marks photos as favorites or adds songs to play lists. Furthermore, items may become popular when many users have consumed them, e.g. by watching a video or a photo, by listening to a song, or clicking on a link. Incorporating different direct and indirect forms of user feedback will allow for a more detailed design of user-item interactions and will decrease the sparsity in the user-item graph.

Additional structuring techniques. Tagging rarely appears in a pure form. Instead, many services will offer additionally structuring functionality, such as sets of predefined categories, groups, or the previously described tag-bundles in Delicious. The quality of tag models could benefit from this additional structure.

Search logs. Yet another valuable source of information are search logs. These logs contain data about the information needs of a user or the entire community. The browsing activity of users that follow the folksonomy graph in order to discover interesting content may additionally help to identify search traces in the graph and to strengthen the relationship between objects.

Even though the potential of these additional information sources is apparent, there currently exists no obvious strategy of how to integrate these sources into existing recommendation models.

Extending the folksonomy graph II

Folksonomy services are not isolated, and external sources may offer valuable additional data about items, users, and tags. Internet databases, such as the Internet Movie Database (IMDb)[12], provide meta-data about films that can help improve the quality of collaborative filtering-based movie recommenders [SCA08]. The DBLP[13] service offers meta-data about publications and authors. Meta-data about books, music, and other content can be retrieved from specialized semantic repositories, such as dbtune.org[14], or universal semantic databases, such as dbpedia[15]. Other external services can be queried in order to derive automated annotations for content, for instance by extracting persons, places, or companies from text, as offered by Reuter's OpenCalais service[16]. External services may also provide additional information about users. For instance, the Friend-of-a-Friend (FOAF) project[17] specifies a machine-readable ontology for describing persons, their activities as well as their relation to other people and objects. Many web services allow the export of user profiles in the FOAF format, such that the activities of users can be traced over more than one service. This additional social data can help in the design of better user models and is especially valuable in cold-start situations, in which new users register to a service. Finally, external information can also enable better tag models. Many authors have shown that the incorporation of external dictionaries, such as WordNet, leads to more accurate and robust tag models, which in turn improve the value of tags as item descriptors. Entity recognition methods could further help to map tags to concepts, likely reducing problems related to tag ambiguity, synonyms, and multilingualism.

[12] http://www.imdb.com/
[13] http://www.informatik.uni-trier.de/~ley/db/index.html
[14] http://dbtune.org/
[15] http://dbpedia.org/
[16] http://www.opencalais.com/
[17] http://www.foaf-project.org/

Tim Berners Lee described the structure that emerges from the interaction of services on the web as "giant global graph" [BL07]. Successful future recommenders will have to move beyond single services and approach *globalization*.

Closing remarks

Tags are only a makeshift solution. Users do not want to tag – they have to tag. Tagging became a success story because of a perceived lack of intelligent information retrieval and management tools. Future solutions that identify the core arguments and entities in all forms of content automatically, that weight the relevance of information with respect to a given context, and provide adaptive information access interfaces will make tagging more and more obsolete. However, these solutions will not materialize overnight. In the meantime, folksonomies can play an important role in learning how users perceive and organize content, and user-generated annotations may act as decisive connectors between content and top-down structuring approaches, paving the way to a *Semantic Web*.

Bibliography

[ABB+09] Fabian Abel, Matteo Baldoni, Cristina Baroglio, Nicola Henze, Daniel Krause, and Viviana Patti. Context-based ranking in folksonomies. In *HT '09: Proc. of the 20th ACM conf. on Hypertext and Hypermedia*. ACM, 2009.

[ACH+04] Einat Amitay, David Carmel, Michael Herscovici, Ronny Lempel, and Aya Soffer. Trend detection through temporal link analysis. *J. of the Am. Soc. for Information Science and Technology*, 55(14), 2004.

[AGMP+07] J. Arenas-García, A. Meng, K. B. Petersen, T. L. Schiøler, L. K. Hansen, and J. Larsen. Unveiling music structure via PLSA similarity fusion. In *IEEE Int. Workshop on Machine Learning for Signal Processing*. IEEE Press, 2007.

[AHDBV05] J. Ignacio Alvarez-Hamelin, Luca Dall'Asta, Alain Barrat, and Alessandro Vespignani. Large scale networks fingerprinting and visualization using the k-core decomposition. In *NIPS '05: Advances in Neural Information Processing Systems*, 2005.

[ALMP08] Ralitsa Angelova, Marek Lipczak, Evangelos Milios, and Pawel Pralat. Characterizing a social bookmarking and tagging network. In *Proc. of the ECAI 2008 Workshop on Mining Social Data (MSoDa)*, 2008.

[ALTY08] Nitin Agarwal, Huan Liu, Lei Tang, and Philip S. Yu. Identifying the influential bloggers in a community. In *WSDM '08: Proc. Int. Conf. on Web search and data mining*. ACM, 2008.

[AN07] Morgan Ames and Mor Naaman. Why we tag: motivations for annotation in mobile and online media. In *CHI '07: Proc. of the SIGCHI conf. on Human factors in computing systems*. ACM, 2007.

[AS08] Rabeeh Abbasi and Steffen Staab. Introducing triple play for improved resource retrieval in collaborative tagging systems. In *In: Proc. of ECIR'08 Workshop on Exploiting Semantic Annotations in IR (ESAIR 2008)*, 2008.

[AT05] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, 17(6), 2005.

[AYGS09] Ching-man Au Yeung, Nicholas Gibbins, and Nigel Shadbolt. Contextualising tags in collaborative tagging systems. In *HT '09: Proc. of the 20th ACM conf. on Hypertext and Hypermedia*. ACM, 2009.

[BB03] Albert-László Barabási and E. Bonabeau. Scale-free networks. *Scientific American*, 288(60-69), 2003.

[BBVW06] Klaus Berberich, Srikanta Bedathur, Michalis Vazirgiannis, and Gerhard Weikum. Buzzrank $...$ and the trend is your friend. In *WWW '06: Proc. of the 15th int. conf. on World Wide Web*. ACM, 2006.

[BCT02] Thorsten Brants, Francine Chen, and Ioannis Tsochantaridis. Topic-based document segmentation with probabilistic latent semantic analysis. In *CIKM '02: Proc. of the eleventh int. conf. on Information and knowledge management*. ACM, 2002.

[BFK01] Zhiqiang Bi, Christos Faloutsos, and Flip Korn. The "dgx" distribution for mining massive, skewed data. In *KDD '01: Proc. of the seventh ACM SIGKDD int. conf. on Knowledge discovery and data mining*. ACM, 2001.

[BFNP08] Kerstin Bischoff, Claudiu S. Firan, Wolfgang Nejdl, and Raluca Paiu. Can all tags be used for search? In *CIKM '08: Proc. of the 17th ACM conf. on Information and knowledge management*. ACM, 2008.

[Bis07] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 1st edition, 2007.

[BL07] Tim Berners-Lee. Giant global graph. online, 2007. http://dig.csail.mit.edu/breadcrumbs/node/215.

[BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3, 2003.

[BP98] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN Systems*, 1998.

[BR04] Belaa Bollobas and Oliver Riordan. The diameter of a scale-free random graph. *Combinatorica*, 24(1), 2004.

[Bra03] Matthew Brand. Fast online svd revisions for lightweight recommender systems. In *SIAM '03: Proc. of the Third SIAM Int. Conf. on Data Mining*, 2003.

[BS97] Marko Balabanović and Yoav Shoham. Fab: content-based, collaborative recommendation. *Commun. ACM*, 40(3), 1997.

[BvdB08] Toine Bogers and Antal van den Bosch. Recommending scientific articles using citeulike. In *RecSys '08: Proc. of the 2008 ACM conf. on Recommender systems*. ACM, 2008.

[BXW+07] Shenghua Bao, Guirong Xue, Xiaoyuan Wu, Yong Yu, Ben Fei, and Zhong Su. Optimizing web search using social annotations. In *WWW '07: Proc. of the 16th int. conf. on World Wide Web*. ACM, 2007.

[Cat06] Ciro Cattuto. Semiotic dynamics in online social communities. *Eur. Phys. J. C*, 46(s02), 2006.

[CBC08] Mark J. Carman, Mark Baillie, and Fabio Crestani. Tag data and personalized ir. In *SSM '08: Proc. of the 2008 ACM workshop on Search in social media*. ACM, 2008.

[CBGC09] Mark J. Carman, Mark Baillie, Robert Gwadera, and Fabio Crestani. A statistical comparison of tag and query logs. In *SIGIR '09: Proc. of the 32nd int. ACM SIGIR conf. on Research and development in IR*. ACM, 2009.

[CBHS08] Ciro Cattuto, Dominik Benz, Andreas Hotho, and Gerd Stumme. Semantic grounding of tag relatedness in social bookmarking systems. In *ISWC '08: Proc. of the 7th Int. Conf. on The Semantic Web*. Springer-Verlag, 2008.

[CBSL07] Ciro Cattuto, Andrea Baldassarri, Vito D. P. Servedio, and Vittorio Loreto. Emergent community structure in social tagging systems. *Advances in Complex Systems*, 11, 2007.

[CF06] Deepayan Chakrabarti and Christos Faloutsos. Graph mining: Laws, generators, and algorithms. *ACM Comput. Surv.*, 38(1), 2006.

[CG96] Stanley F. Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proc. of the 34th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 1996.

[CH00] David A. Cohn and Thomas Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. In *NIPS*. MIT Press, 2000.

[CLP07] Ciro Cattuto, Vittorio Loreto, and Luciano Pietronero. Collaborative tagging and semiotic dynamics. *PNAS*, 104, 2007.

[CM08] Ed H. Chi and Todd Mytkowicz. Understanding the efficiency of social tagging systems using information theory. In *HT '08: Proc. of the nineteenth ACM conf. on Hypertext and hypermedia*. ACM, 2008.

[CMG09] Meeyoung Cha, Alan Mislove, and Krishna P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. In *WWW '09: Proc. of the 18th int. conf. on World wide web*. ACM, 2009.

[CSB+07] Ciro Cattuto, Christoph Schmitz, Andre Baldassarri, Vito D. P. Servedio, Vittorio Loreto, Andreas Hotho, Miranda Grahl, and Gerd Stumme. Network properties of folksonomies. *AI Communications Special Issue on "Network Analysis in Natural Sciences and Engineering"*, 2007.

[CTT06] Yun Chi, Belle L. Tseng, and Junichi Tatemura. Eigen-trend: trend analysis in the blogosphere based on singular value decompositions. In *CIKM '06: Proc. of the 15th ACM conf. on Information and knowledge management*. ACM, 2006.

[DDF+90] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *J. of the Am. Soc. for Information Science*, 41, 1990.

[DDGR07] Abhinandan Das, Mayur Datar, Ashutosh Garg, and Shyam-Sundar Rajaram. Google news personalization: scalable online collaborative filtering. In Carey L. Williamson, Mary Ellen Zurko, Peter F. Patel-Schneider, and Prashant J. Shenoy, editors, *WWW*. ACM, 2007.

[Del07a] Delicious. delicious is four today. online, 2007. http://blog.delicious.com/blog/2007/09/delicious-is-fo.html.

[Del07b] Delicious. Sharing made easier: Email and tweet your bookmarks. online, 2007. http://blog.delicious.com/blog/2009/08/sharing-made-easier-email-and-tweet-your-bookmarks.html.

[dGLSB08] Marco de Gemmis, Pasquale Lops, Giovanni Semeraro, and Pierpaolo Basile. Integrating tags in a semantic content-based recommender. In *RecSys '08: Proc. of the 2008 ACM conf. on Recommender systems*. ACM, 2008.

[DKM+06] Micah Dubinko, Ravi Kumar, Joseph Magnani, Jasmine Novak, Prabhakar Raghavan, and Andrew Tomkins. Visualizing tags over time. In *WWW '06: Proc. of the 15th int. conf. on World Wide Web*. ACM Press, 2006.

[DLDMV00] Lieven De Lathauwer, Bart De Moor, and Joos Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl*, 21, 2000.

[DS08] Klaas Dellschaft and Steffen Staab. An epistemic dynamic model for tagging systems. In *HT '08: Proc. of the nineteenth ACM conf. on Hypertext and hypermedia*. ACM, 2008.

[Faw06] Tom Fawcett. An introduction to roc analysis. *Pattern Recognition Letters*, 27(8), 2006.

[Fli05] Flickr. Yahoo actually does acquire flickr. online, 2005. http://blog.flickr.net/en/2005/03/20/yahoo-actually-does-acquire-flickr/.

[FNP07] Claudiu S. Firan, Wolfgang Nejdl, and Raluca Paiu. The benefit of using tag-based profiles. In *LA-WEB '07: Proc. of the 2007 Latin Am. Web Conf.* IEEE Computer Society, 2007.

[GBM+09] Ziyu Guan, Jiajun Bu, Qiaozhu Mei, Chun Chen, and Can Wang. Personalized tag recommendation using graph-based ranking on multi-type interrelated objects. In *SIGIR '09: Proc. of the 32nd int. ACM SIGIR conf. on Research and development in IR*. ACM, 2009.

[GH06] Scott A. Golder and Bernardo A. Huberman. Usage patterns of collaborative tagging systems. *J. of Information Science*, 32(2), 2006.

[GHS07] Miranda Grahl, Andreas Hotho, and Gerd Stumme. Conceptual clustering of social bookmarking sites. In *7th Int. Conf. on Knowledge Management (I-KNOW '07)*. Know-Center, 2007.

[GHT04] Natalie S. Glance, Matthew Hurst, and Takashi Tomokiyo. Blogpulse: Automated trend discovery for weblogs. In *WWW '04: Workshop on the Weblogging Ecosystem*. ACM, 2004.

[GK08] A. Gkanogiannis and T. Kalamboukis. A novel supervised learning algorithm and its use for spam detection in social bookmarking systems. In *RSDC '08: Proc. of the Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Discovery Challenge*, 2008.

[GR70] G. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5), 1970.

[GSRLC09] Jonathan Gemmell, Thomas Schimoler, Maryam Ramezani, and Bamshad Mobasher Laura Christiansen. Improving folkrank with item-based collaborative filtering. In *RecSys: Workshop on Recommender Systems and the Social Web*, 2009.

[GSRM09] Jonathan Gemmell, Thomas Schimoler, Maryam Ramezani, and Bamshad Mobasher. Adapting k-nearest neighbor for tag recommendation in folksonomies. In *Proc. of the Association for the Advancement of Artificial Intelligence (AAAI)*, 2009.

[GW08] Nikhil Garg and Ingmar Weber. Personalized, interactive tag recommendation for flickr. In *RecSys '08: Proc. of the 2008 ACM conf. on Recommender systems*. ACM, 2008.

[Hen09] Leonhard Hennig. Topic-based multi-document summarization with probabilistic latent semantic analysis. In *RANLP '09*, 2009.

[Hey06] Hector Heymann, Paul Garcia-Molina. Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical report, Computer Science Department, Stanford University, 2006.

[HJSS06a] Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme. Ir in folksonomies: Search and ranking. In *Proc. of the 3rd Eur. Semantic Web Conf. (ESWC)*. Springer, 2006.

[HJSS06b] Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme. Trend detection in folksonomies. In *SAMT*, volume 4306 of *Lecture Notes in Computer Science*. Springer, 2006.

[HKGM08] Paul Heymann, Georgia Koutrika, and Hector Garcia-Molina. Can social bookmarking improve web search? In *WSDM '08: Proc. of the int. conf. on Web search and data mining*. ACM, 2008.

[HKTR04] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1), 2004.

[Hof99a] Thomas Hofmann. Learning the similarity of documents: An information-geometric approach to document retrieval and categorization. In *NIPS*. The MIT Press, 1999.

[Hof99b] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proc. of Uncertainty in Artificial Intelligence, UAI'99*, 1999.

[HRGM08] Paul Heymann, Daniel Ramage, and Hector Garcia-Molina. Social tag prediction. In *SIGIR '08: Proc. of the 31st ann. int. ACM SIGIR conf. on Research and development in IR*. ACM, 2008.

[HRS07] Harry Halpin, Valentin Robu, and Hana Shepherd. The complex dynamics of collaborative tagging. In *WWW '07: Proc. of the 16th int. conf. on World Wide Web*. ACM, 2007.

[JEHS09] Robert Jäschke, Folke Eisterlehner, Andreas Hotho, and Gerd Stumme. Testing and evaluating tag recommenders in a live system. In *RecSys '09: Proc. of the 2009 ACM Conf. on Recommender Systems*. ACM, 2009. (to appear).

[JJF08] Akshay Java, Anupam Joshi, and Tim Finin. Detecting commmunities via simultaneous clustering of graphs and folksonomies. In *Proc. of the 10th Workshop on Web Mining and Web Usage Analysis (WebKDD)*. ACM, 2008. Held in conjunction with The 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD 2008).

[JMH+07] Robert Jäschke, Leandro Marinho, Andreas Hotho, Lars Schmidt-Thieme, and Gerd Stumme. Tag recommendations in folksonomies. In *Lernen - Wissensentdeckung - Adaptivität (LWA) Workshop Proc.*, 2007.

[JZM04] Xin Jin, Yanzan Zhou, and Bamshad Mobasher. Web usage mining based on probabilistic latent semantic analysis. In Won Kim, Ron Kohavi, Johannes Gehrke, and William DuMouchel, editors, *KDD '04: Proc. of the 15th ACM SIGKDD int. conf. on Knowledge discovery and data mining*. ACM, 2004.

[KEG+07] Georgia Koutrika, Frans Adjie Effendi, Zoltán Gyöngyi, Paul Heymann, and Hector Garcia-Molina. Combating spam in tagging systems. In *AIRWeb '07: Proc. of the 3rd int. workshop on Adversarial inf. retrieval on the web*, 2007.

[KF09] Ralf Krestel and Peter Fankhauser. Latent dirichlet allocation for tag recommendation. In *RecSys '09: Proc. of the 2008 ACM conf. on Recommender systems*. ACM, 2009.

[KHS08a] Beate Krause, Andreas Hotho, and Gerd Stumme. The anti-social tagger - detecting spam in social bookmarking systems. In *Proc. of the 4th Int. Workshop on Adversarial IR on the Web*, 2008.

[KHS08b] Beate Krause, Andreas Hotho, and Gerd Stumme. A comparison of social bookmarking with traditional search. In *Advances in IR, 30th Eur. Conf. on IR Research, ECIR 2008*, volume 4956. Springer, 2008.

[KJHS08] Beate Krause, Robert Jäschke, Andreas Hotho, and Gerd Stumme. Logsonomy - social ir with logdata. In *HT '08: Proc. of the nineteenth ACM conf. on Hypertext and hypermedia*. ACM, 2008.

[Kle99] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. In *J. of the ACM*, volume 46. ACM, 1999.

[Kle06] J. Kleinberg. Temporal dynamics of on-line information streams. In *Data Stream Management: Processing High-Speed Data Streams*. Springer, 2006.

[Kor09] Yehuda Koren. Collaborative filtering with temporal dynamics. In *KDD '09: Proc. of the 15th ACM SIGKDD int. conf. on Knowledge discovery and data mining*, 2009.

[Kro05] Ellyssa Kroski. The hive mind: Folksonomies and user-based tagging. online, 2005. http://infotangle.blogsome.com/2005/12/07/the-hive-mind-folksonomies-and-user-based-tagging/.

[KS08] Tamara G. Kolda and Jimeng Sun. Scalable tensor decompositions for multi-aspect data mining. In *ICDM 2008: Proc. of the 8th IEEE Int. Conf. on Data Mining*, 2008.

[LAH07] Jure Leskovec, Lada A. Adamic, and Bernardo A. Huberman. The dynamics of viral marketing. *ACM Trans. Web*, 1(1), 2007.

[LEC07] David Laniado, Davide Eynard, and Marco Colombetti. Using wordnet to turn a folksonomy into a hierarchy of concepts. In *SWAP '07: Proc. of the 4th Italian Semantic Web Workshop*, 2007.

[LGZ08] Xin Li, Lei Guo, and Yihong Eric Zhao. Tag-based social interest discovery. In *WWW '08: Proc. of the 17th int. conf. on World Wide Web*. ACM, 2008.

[LH08] Jure Leskovec and Eric Horvitz. Planetary-scale views on a large instant-messaging network. In *WWW '08: Proc. of the 17th int. conf. on World Wide Web*. ACM, 2008.

[LHY+09] Dong Liu, Xian-Sheng Hua, Linjun Yang, Meng Wang, and Hong-Jiang Zhang. Tag ranking. In *WWW '09: Proc. of the 18th int. conf. on World wide web*. ACM, 2009.

[Lip08] Marek Lipczak. Tag recommendation for folksonomies oriented towards individual users. In *RSDC '08: Proc. of the Eur. Conf. on Machine Learning and Principles and Practice* *of Knowledge Discovery in Databases (ECML/PKDD), Discovery Challenge*, 2008.

[LYHM09] Marek Lipczak, Yael Kollet Yeming Hu, and Evangelos Milios. Tag sources for recommendation in collaborative tagging systems. In *Proc. of the Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Discovery Challenge*, 2009.

[Mat04] Adam Mathes. Folksonomies - cooperative classification and communication through shared metadata. online, 2004. http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html.

[mAYGS08] Ching man Au Yeung, Nicholas Gibbins, and Nigel Shadbolt. A study of user profile generation from folksonomies. In *WWW '08: Social Web Workshop*, 2008.

[MBA+09] Johannes Mrosek, Stefan Bussmann, Hendrik Albers, Kai Posdziech, Benedikt Hengefeld, Nils Opperman, Stefan Robert, and Gerrit Spira. Content- and graph-based tag recommendation: Two variations. In *Proc. of the Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Discovery Challenge*, 2009.

[MCM+09] 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: Proc. of the 18th int. conf. on World Wide Web*. ACM, 2009.

[MGP03] Florent Monay and Daniel Gatica-Perez. On image auto-annotation with latent space models. In *MULTIMEDIA '03: Proc. of the eleventh ACM int. conf. on Multimedia*. ACM, 2003.

[Mik05] Peter Mika. Ontologies are us: A unified model of social networks and semantics. *J. Web Sem.*, 5(1), 2005.

[MMN02] Prem Melville, Raymod J. Mooney, and Ramadass Nagarajan. Content-boosted collaborative filtering for improved recommendations. In *18th nat. conf. on Artificial intelligence*. Am. Ass. for Artificial Intelligence, 2002.

[MNBD06] Cameron Marlow, Mor Naaman, Danah Boyd, and Marc Davis. Ht06, tagging paper, taxonomy, flickr, academic article, to read. In *HT '06*. ACM, 2006.

[MR09] Tariq Mahmood and Francesco Ricci. Improving recommender systems with adaptive conversational strategies. In *HT '09: Proc. of the 20th ACM conf. on Hypertext and hypermedia*. ACM, 2009.

[MRS09] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *An Introduction to Information Retrieval*. Cambridge University Press, Cambridge, England, 2009.

[MS99] Chris Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[NAYG+09] Michael G. Noll, Ching-man Au Yeung, Nicholas Gibbins, Christoph Meinel, and Nigel Shadbolt. Telling experts from spammers: expertise ranking in folksonomies. In *SIGIR '09: Proc. of the 32nd int. ACM SIGIR conf. on Research and development in IR*. ACM, 2009.

[New03] Mark E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45, 2003.

[NM07] Michael Noll and Christoph Meinel. Web search personalization via social bookmarking and tagging. In *Proc. of the 6th Int. Semantic Web Conf. and 2nd Asian Sem. Web Conf. (ISWC/ASWC2007)*. Springer, 2007.

[NO08] Nicolas Neubauer and Klaus Obermayer. Predicting tag spam examining cooccurrences, network structures and url components. In *RSDC '08: Proc. of the Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Discovery Challenge*, 2008.

[NO09] Nicolas Neubauer and Klaus Obermayer. Hyperincident connected components of tagging networks. In *HT '09: Proc. of the 20th ACM conf. on Hypertext and hypermedia*. ACM, 2009.

[NPY+09] Tae-Gil Noh, Seong-Bae Park, Hee-Geun Yoon, Sang-Jo Lee, and Se-Young Park. An automatic translation of tags for multimedia contents using folksonomy networks. In *SIGIR '09:*

Proc. of the 32nd int. ACM SIGIR conf. on Research and development in IR. ACM, 2009.

[NWO09] Nicolas Neubauer, Robert Wetzker, and Klaus Obermayer. Tag spam creates large non-giant connected components. In *AIRWeb '09: Proc. of the 5th Int. Workshop on Adversarial IR on the Web*. ACM, 2009.

[OGC08] Steffen Oldenburg, Martin Garbe, and Clemens Cap. Similarity cross-analysis of tag / co-tag spaces in social classification systems. In *SSM '08: Proc. of the 2008 ACM workshop on Search in social media*. ACM, 2008.

[O'R05] Tim O'Reilly. What is web 2.0 - design patterns and business models for the next generation of software. online, 2005. http://oreilly.com/web2/archive/what-is-web-20.html.

[OSvZ09] Simon Overell, Börkur Sigurbjörnsson, and Roelof van Zwol. Classifying tags using open content resources. In *WSDM '09: Proc. of the Second ACM Int. Conf. on Web Search and Data Mining*. ACM, 2009.

[PUPL01] Alexandrin Popescul, Lyle H. Ungar, David M. Pennock, and Steve Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *UAI '01: Proc. of the 17th Conf. in Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 2001.

[Qui05] Emanuele Quintarelli. Folksonomies: power to the people. online, 2005. http://www.iskoi.org/doc/folksonomies.htm.

[RBMNST09] Steffen Rendle, Leandro Balby Marinho, Alexandros Nanopoulos, and Lars Schmidt-Thieme. Learning optimal ranking with tensor factorization for tag recommendation. In *KDD '09: Proc. of the 15th ACM SIGKDD int. conf. on Knowledge discovery and data mining*. ACM, 2009.

[RHMGM09] Daniel Ramage, Paul Heymann, Christopher D. Manning, and Hector Garcia-Molina. Clustering the tagged web. In *WSDM '09: Proc. of the Second ACM Int. Conf. on Web Search and Data Mining*. ACM, 2009.

[RST09] Steffen Rendle and Lars Schmidt-Thieme. Factor models for tag recommendation in bibsonomy. In *ECML/PKDD '09: Proc. of the Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2009.

[RV97] Paul Resnick and Hal R. Varian. Recommender systems. *Commun. ACM*, 40(3), 1997.

[SCA+07] Martin Szomszor, Ciro Cattuto, Harith Alani, Kieron O'Hara, Andrea Baldassarri, Vittorio Loreto, and Vito D. P. Servedio. Folksonomies, the semantic web, and movie recommendation. In *Bridging the Gap between Semantic Web and Web 2.0 (SemNet 2007)*, 2007.

[SCA08] Martin N. Szomszor, Iván Cantador, and Harith Alani. Correlating user profiles from multiple folksonomies. In *HT '08: Proc. of the nineteenth ACM conf. on Hypertext and hypermedia*. ACM, 2008.

[SGMB08] Andriy Shepitsen, Jonathan Gemmell, Bamshad Mobasher, and Robin Burke. Personalized recommendation in social tagging systems using hierarchical clustering. In *RecSys '08: Proc. of the 2008 ACM conf. on Recommender systems*. ACM, 2008.

[Shi05] Clay Shirky. Ontology is overrated: Categories, links, and tags. online, 2005. http://www.shirky.com/writings/ontology_overrated.html.

[SHLR07] Shilad Sen, F. Maxwell Harper, Adam LaPitz, and John Riedl. The quest for quality tags. In *GROUP '07: Proc. of the 2007 int. ACM conf. on Supporting group work*. ACM, 2007.

[SKKR00] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl. Application of dimensionality reduction in recommender system - a case study. In *ACM WebKDD Workshop*, 2000.

[SKKR01] Badrul Sarwar, George Karypis, Joseph Konstan, and John Reidl. Item-based collaborative filtering recommendation algorithms. In *WWW '01: Proc. of the 10th int. conf. on World Wide Web*. ACM, 2001.

[SKKR02] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Incremental singular value decomposition algorithms for highly scalable recommender systems. In *Fifth Int. Conf. on Computer and Information Science*, 2002.

[SLR+06] Shilad Sen, Shyong K. Lam, Al Mamunur Rashid, Dan Cosley, Dan Frankowski, Jeremy Osterhouse, F. Maxwell Harper, and John Riedl. tagging, communities, vocabulary, evolution. In *CSCW '06: Proc. of the 2006 20th anniversary conf. on Computer supported cooperative work*. ACM, 2006.

[SNCA+09] Elizeu Santos-Neto, David Condon, Nazareno Andrade, Adriana Iamnitchi, and Matei Ripeanu. Individual and social behavior in tagging systems. In *HT '09: Proc. of the 20th ACM conf. on Hypertext and hypermedia*. ACM, 2009.

[SNM08] Panagiotis Symeonidis, Alexandros Nanopoulos, and Yannis Manolopoulos. Tag recommendations based on tensor dimensionality reduction. In *RecSys '08: Proc. of the 2008 ACM conf. on Recommender systems*. ACM, 2008.

[SPUP02] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock. Methods and metrics for cold-start recommendations. In *SIGIR '02: Proc. of the 25th ann. int. ACM SIGIR conf. on Research and development in IR*. ACM, 2002.

[SS09] Stefan Siersdorfer and Sergej Sizov. Social recommender systems for web 2.0 folksonomies. In *HT '09: Proc. of the 20th ACM conf. on Hypertext and hypermedia*. ACM, 2009.

[SSH97] Bracha Shapira, Peretz Shoval, and Uri Hanani. Stereotypes in information filtering systems. *Information Processing & Management*, 33(3), 1997.

[Str08] Markus Strohmaier. Purpose tagging: capturing user intent to assist goal-oriented social search. In *SSM '08: Proc. of the 2008 ACM workshop on Search in social media*. ACM, 2008.

[Sur04] James Surowiecki. *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations*. Doubleday, 2004.

[SVG08] Fabian M. Suchanek, Milan Vojnovic, and Dinan Gunawardena. Social tags: meaning and suggestions. In *CIKM '08: Proc. of the 17th ACM conf. on Information and knowledge management*. ACM, 2008.

[SVR09] Shilad Sen, Jesse Vig, and John Riedl. Tagommenders: connecting users to items through tags. In *WWW '09: Proc. of the 18th int. conf. on World Wide Web*. ACM, 2009.

[SvZ08] Börkur Sigurbj¨örnsson and Roelof van Zwol. Flickr tag recommendation based on collective knowledge. In *WWW '08: Proc. of the 17th int. conf. on World Wide Web*. ACM, 2008.

[SWUH09] Alan Said, Robert Wetzker, Winfried Umbrath, and Leonhard Hennig. A hybrid plsa approach for warmer cold start in folksonomy recommendation. In *RecSys '09: Workshop on Recommender Systems and the Social Web*, 2009.

[SZG08] Yang Song, Lu Zhang, and C. Lee Giles. A sparse gaussian processes classification framework for fast tag suggestions. In *CIKM '08: Proc. of the 17th ACM conf. on Information and knowledge management*. ACM, 2008.

[SZL+08] Yang Song, Ziming Zhuang, Huajing Li, Qiankun Zhao, Jia Li, Wang-Chien Lee, and C. Lee Giles. Real-time automatic tag recommendation. In *SIGIR '08: Proc. of the 31st int. ACM SIGIR conf. on Research and development in IR*. ACM, 2008.

[TSD08] M. Tatu, M. Srikanth, and T. D'Silva. Rsdc '08: Tag recommendations using bookmark content. In *RSDC '08: Proc. of the Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Discovery Challenge*, 2008.

[TSMT08] Karen H. L. Tso Sutter, Leandro B. Marinho, and Lars S. Thieme. Tag-aware recommender systems by fusion of collaborative filtering algorithms. In *SAC '08: Proc. of the 2008 ACM symposium on Applied computing*. ACM, 2008.

[Vos06] Jakob Voss. Collaborative thesaurus tagging the wikipedia way. *CoRR*, abs/cs/0604036, 2006.

[vZMGPR08] Roelof van Zwol, Vanessa Murdock, Lluis Garcia Pueyo, and Georgina Ramirez. Diversifying image search with user generated content. In *MIR '08: Proc. of the 1st ACM int. conf. on Multimedia IR*. ACM, 2008.

[Wal04] Thomas Vander Wal. Folksonomy coinage and definition. online, 2004. http://www.vanderwal.net/folksonomy.html.

[Wal05] Thomas Vander Wal. Explaining and showing broad and narrow folksonomies. online, 2005. http://www.vanderwal.net/random/entrysel.php?blog=1635.

[WPK+08] Robert Wetzker, Till Plumbaum, Alexander Korth, Christian Bauckhage, Tansu Alpcan, and Florian Metze. Detecting trends in social bookmarking systems using a probabilistic generative model and smoothing. In *Proc. of the Int. Conf. on Pattern Recognition (ICPR)*, 2008.

[WS98] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684), 1998.

[WSVZ08] Kilian Quirin Weinberger, Malcolm Slaney, and Roelof Van Zwol. Resolving tag ambiguity. In *MM '08: Proc. of the 16th ACM int. conf. on Multimedia*. ACM, 2008.

[WSZ09] Robert Wetzker, Alan Said, and Carsten Zimmermann. Understanding the user: Personomy translation for tag recommendation. In *ECML/PKDD '09: Proc. of the Eur. Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, 2009.

[WUS09] Robert Wetzker, Winfried Umbrath, and Alan Said. A hybrid approach to item recommendation in folksonomies. In *ESAIR '09: Proc. of the WSDM '09 Workshop on Exploiting Semantic Annotations in IR*. ACM, 2009.

[WYYH09] Lei Wu, Linjun Yang, Nenghai Yu, and Xian-Sheng Hua. Learning to tag. In *WWW '09: Proc. of the 18th int. conf. on World wide web*. ACM, 2009.

[WZB08] Robert Wetzker, Carsten Zimmermann, and Christian Bauckhage. Analyzing social bookmarking systems: A del.icio.us cookbook. In *MSoDa '08: Proc. of the ECAI '08 Workshop on Mining Social Data*, 2008.

[WZB10a] Robert Wetzker, Carsten Zimmermann, and Christian Bauckhage. Detecting trends in social bookmarking systems: A del.icio.us endeavor. *Int. Journal of Data Warehousing and Mining*, 2010. (to appear).

[WZB10b] Robert Wetzker, Carsten Zimmermann, and Christian Bauckhage. I tag, you tag: Personomy translation for advanced user models. In *WSDM '10: Proc. Int. Conf. on Web search and data mining*, 2010.

[WZY06] Xian Wu, Lei Zhang, and Yong Yu. Exploring social annotations for the semantic web. In *WWW '06: Proc. of the 15th int. conf. on World Wide Web*. ACM, 2006.

[XBF+08] Shengliang Xu, Shenghua Bao, Ben Fei, Zhong Su, and Yong Yu. Exploring folksonomy for personalized search. In *SIGIR*. ACM, 2008.

[XFF+05] Wensi Xi, Edward A. Fox, Weiguo Fan, Benyu Zhang, Zheng Chen, Jun Yan, and Dong Zhuang. Simfusion: measuring similarity using unified relationship matrix. In *SIGIR '05: Proc. of the 28th annual int. ACM SIGIR conf. on Research and development in IR*. ACM, 2005.

[XFMS06] Zhichen Xu, Yun Fu, Jianchang Mao, and Difu Su. Towards the semantic web: Collaborative tag suggestions. In *Proc. of Collaborative Web Tagging Workshop at 15th Int. World Wide Web Conf.*, 2006.

[YBLS08] Sihem Amer Yahia, Michael Benedikt, Laks V. S. Lakshmanan, and Julia Stoyanovich. Efficient network aware search in collaborative tagging sites. *Proc. VLDB Endow.*, 1(1), 2008.

[YJNT07] Yusuke Yanbe, Adam Jatowt, Satoshi Nakamura, and Katsumi Tanaka. Can social bookmarking enhance search in the web? In *JCDL '07: Proc. of the 7th ACM/IEEE-CS joint conf. on Digital libraries*. ACM, 2007.

[YLMH09] Zhijun Yin, Rui Li, Qiaozhu Mei, and Jiawei Han. Exploring social tagging graph for web object classification. In *KDD '09: Proc. of the 15th ACM SIGKDD int. conf. on Knowledge discovery and data mining*. ACM, 2009.

[Zaw05] Jeremy Zawodny. Great tastes that go great together. online, 2005. http://www.ysearchblog.com/2005/12/09/great-tastes-that-go-great-together/.

[ZBZ+08] Ding Zhou, Jiang Bian, Shuyi Zheng, Hongyuan Zha, and C. Lee Giles. Exploring social annotations for ir. In *WWW '08: Proc. of the 17th int. conf. on World Wide Web*. ACM, 2008.

[ZC08] Valentina Zanardi and Licia Capra. Social ranking: uncovering relevant content using tag-based recommender systems. In *RecSys '08: Proc. of the 2008 ACM conf. on Recommender systems*. ACM, 2008.

[Zip49] George K. Zipf. *Human Behavior and the Principle of Least Effort*. Addison-Wesley (Reading MA), 1949.

[ZML09] Dell Zhang, Robert Mao, and Wei Li. The recurrence dynamics of social tagging. In *WWW '09: Proc. of the 18th int. conf. on World wide web*. ACM, 2009.

[ZWY06] Lei Zhang, Xian Wu, and Yong Yu. Emergent semantics from folksonomies: A quantitative study. *J. Data Semantics VI*, 4090, 2006.