A Socially Intelligent Approach for Enterprise Information Search and Recommendation

Konstantinos Christidis\(^1\), Dimitris Apostolou\(^2\), Gregoris Mentzas\(^1\)

\(^1\)National Technical University of Athens, \{kchrist, gmentzas\}@mail.ntua.gr
\(^2\)Department of Informatics, University of Piraeus, dapost@unipi.gr

Abstract
This paper focuses on the development of socially intelligent computing systems at the enterprise level. Specifically, in order to improve the information search and recommendation functionalities of social business software, we extend corporate knowledge structuring approaches, such as folksonomies and taxonomies, with the addition of statistical topic models. We use probabilistic models in order to uncover hidden topics in the corporate ‘knowledge base’ and hence add an intelligent perspective in social collaboration. Probabilistic topic models are based upon the idea that documents are mixtures of topics, where a topic is defined as a probability distribution over words. We apply our approach in the Organik social business software platform and deploy it in five companies. Our results showed enhanced recommendations and improved search efficiency, while our approach effectively addresses problems in query expansion and recommends relevant resources and tags which in turn can leverage the creation and evolution of social knowledge structures like folksonomies.

Keywords

1 Introduction

Socially intelligent computing systems have recently emerged as a new interdisciplinary research area. In socially intelligent computing systems “social refers to the interactions among people and increasingly more sophisticated computing technologies; intelligent refers to the emerging intelligence exhibited by such systems as well as their increasing knowledge about people and their interactions with one another and with computers; computing refers to the computation technologies that act as mediators among people, as tools used by people, and as equal or complementary participants with people” [Maher, 2009], [Agarwal and Xu, 2011]. Research in socially intelligent computing broadens the scope and coverage of social computing [Wang, et al. 2005] and spans a variety of issues ranging from intelligent collaborative systems [Prinz, et al. 2010] to the use of semantic technologies and the web of linked open data in order to support computational sociality [Sheth and Nagarajan 2009], while related applications use intelligent computing infrastructures that exploit social media on the web [Bothos, et al. 2010].

At the business level, socially intelligent computing systems can be considered the intelligent evolution of social business software that facilitate Enterprise 2.0 companies as defined by [McAfee 2006]. Enterprise 2.0, and the supporting social software, is based largely on fostering internal and external communities using social, participatory Web tools and aligning them with the business needs. Enterprise 2.0 is already a reality in numerous companies around the globe; see e.g. [Kaiser et al. 2009]. Further exploiting technologies like semantic web and linked open data approaches, or generally Web 3.0 technologies [Hendler, 2009] [Hendler, 2010], [Breslin et al. 2010] can pave the way towards socially intelligent business; see Figure 1.

A critical challenge in socially intelligent business is to discover and recommend useful information resources within the increased ‘knowledge base’ of the organisation. To facilitate this task, categorization of resources with knowledge structures is commonly used.
Knowledge structures range from the most simple and least expressive, such as folksonomies, to the most complex and precise ones, such as ontologies. Folksonomies are bottom-up catalogues of tags. Ontologies are machine-readable specifications of domain knowledge [Uschold and Gruninger, 2004]. Taxonomies, simple hierarchies of terms, are also used extensively in social business software.

In this paper we focus on exploiting knowledge structures in social business software in order to enhance search and recommendation of appropriate information resources. In particular we investigate approaches that use probabilistic topic models [Blei 2003] in order to uncover hidden topics in the organisational ‘knowledge base’ and add an intelligent perspective in social collaboration at the enterprise level [Chi et al 2011]. Specifically we extend our previous work [Christidis and Mentzas, 2010] and test our approach by embedding our algorithms in the Organik social business software [Christidis et al. 2011]. We deployed Organik in five knowledge-intensive companies. Our results show enhanced recommendations and improved search efficiency; our approach effectively addresses problems in query expansion and recommends relevant resources and tags which in turn can leverage the creation and evolution of social knowledge structures like folksonomies.

The remainder of this paper is organized as follows. The following section gives an introduction to information search and recommender systems as well as an overview of the relevant knowledge structures used. In section 3 we describe probabilistic topic models, the family of information processing techniques that we employed for uncovering latent topics in corporate knowledge bases and provide a description of our proposal that uses latent topics to provide intelligence in search and recommendation. Section 4 describes our case study and an evaluation of our approach in real-life settings while section 5 gives an overview of related work. Conclusions and further work are provided in the final section.

2 Knowledge Structures in Information Search and Recommendation

Information search refers to “finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections” [Manning, 2008]. Search refers to any interface that allows a number of user-provoked information retrieval tasks. In the past fifty years, search has evolved from data search to information search, to syntax search, and lately to semantic search [Baeza – Yates et al. 1999]. On the other hand, recommender systems are addressing the problem of estimating the utility or the ratings of items that have not yet been seen by the user. To address this problem different types of recommender systems have been researched, content-based or collaborative, utilizing models or heuristics.
Business interest in this area started from early electronic businesses techniques, such as the collaborative filtering recommendations of Amazon.com. In addition to e-business, recommender systems are used by organisational Knowledge Management Systems (KMS) where the problem is redefined as estimating the relevance and usefulness of a previously unseen information resource (e.g., a document, an expert profile, a discussion thread) [Maier 2007].

Search and recommendation technologies are not expected to work solely with unstructured information. The usage of knowledge structures provides some amount of categorization and structure. Both the choice of the software platform and the approach adopted by the organisation determine the degree of formality in knowledge structures. Knowledge structures can improve the processes of both search and recommenders. In recommenders, elements such as concepts or tags can be used for calculating relevance; in search, the existence of a concept hierarchy can enable searching for broader/narrower or similar terms.

Classes of information resources are modelled using various types of knowledge structures. In contemporary knowledge management systems the problem of generating domain knowledge structures has been tackled with the use of corporate taxonomies i.e., classification schemes organising domain entities in hierarchical tree structures [Gilchrist 2001]. Taxonomies are typically engineered in a top-down approach where the subject area is divided into increasingly narrower and more detailed systematically enumerated categories. However, social business software is based primarily on the use of folksonomies, i.e., collaborative, user-generated metadata that offer an informal way of information categorisation. Folksonomies are created bottom-up, in a way where the subject area is divided into individual concepts which can be composed to construct complex subjects via appropriate sets of rules [Dotsika 2009].

Limitations and challenges in the application of knowledge structures combined with the increasing volume of user generated information in enterprise social software can lead to an inferior quality of search and recommendation results [Agichtein et al. 2008]. We therefore need new ways to reveal the ever changing knowledge of the domain and to map it to the knowledge structures.

Our motivation for the work presented in this paper is to build intelligence within the search and recommendation functionalities of social business software. To this end, we utilise statistical methods for supporting search and recommendation that work in tandem with both folksonomy- and taxonomy-based knowledge structures. We build a topic model of the knowledge present in the platform and utilise it in order to improve search, provide recommendations for resources and tags, and thus support socially intelligent computational systems at the enterprise level.

3 Intelligent Extensions to Information Discovery and Recommendation

3.1 Probabilistic Topic Models

In order to improve the information resource discovery and recommendation functionalities of social business software, we propose the extension of knowledge structuring approaches, such as folksonomies and taxonomies, with the addition of latent topics. We use probabilistic topic models as a technical background for uncovering these latent topics. Probabilistic topic models are based upon the idea that documents are mixtures of topics, where a topic is defined as a probability distribution over words. Statistical methods can be used in order to discover a model that describes the way by which documents can be generated [Steyvers 2007]. Specifically we use Latent Dirichlet Allocation (LDA) [Blei 2003] to elicit latent topics and use them to identify similarities; similarities are then used for information resource recommendation and for the expansion of query results.

LDA is a generative method, since it is based on the assumption that a generative model can sufficiently describe the document corpus and this model can be discovered. The model uses
latent topic variables and probabilistic sampling techniques to generate the documents: documents are produced by sampling topics from a distribution of topics over documents and sampling words from a distribution of words over each topic. To fully describe the model these distributions need to be identified. To this end, LDA is trained and distributions converge before the model can be used in applications.

Figure 2 illustrates in plate notation the generative model: \( z \) and \( d \) variables identify topics and documents, while \( \theta(d) \) is the distribution over topics for a document \( d \) and \( \phi(z) \) is the distribution over words for a topic \( z \). These distributions can be used to generate documents in the form of a collection of words \( (w) \). \( D \) is the number of documents, \( T \) is the number of topics in the corpus and \( N_d \) the topics found in each document. Hyper-parameters \( \alpha \) and \( \beta \) identify the Dirichlet priors of the above multinomial distributions respectively. These hyper-parameters can be changed in order to control the smoothing of the distributions.

Instead of directly estimating the two required distributions, \( \theta \) and \( \phi \), it is advisable to estimate directly the posterior distribution over \( z \), that is, assignment of word tokens to topics. A Gibbs sampler, which is a special case of a Monte Carlo Markov Chain, is used for this approximation of \( p(z) \), which subsequently is used to estimate \( \phi \) and \( \theta \). Iterative evaluation of (1), after a burn-in period, leads to a sampling convergence to an estimate of \( z \). Then using (2) the topic-word and document-topic distributions can be calculated. \( C^WT \) and \( C^DT \) are matrices of counts: 

\[
C^WT_{wj} \text{ contains the number of times word } w \text{ is assigned to topic } j, \text{ not including the current instance } i \\
C^DT_{dj} \text{ contains the number of times topic } j \text{ is assigned to some word token in document } d, \text{ not including the current instance } i.
\]

LDA exhibits qualities and drawbacks. One of the main advantages of the method is that it is naturally generalized to new documents. After the topics have been trained, it is possible to infer the distribution that could have generated a new, previously unseen, item. Additionally the parameters needed are not growing with the size of the training corpus. The topics generated by this method are not epistemologically claimed to be more than latent multinomial variables, nevertheless are capturing probability distribution of words based on the co-occurrence. Moreover, these distributions are exchangeable, i.e. after the document re-training no assumptions can be made to relate topics from the previous with topics from the current model.
3.2 Enhancing Search and Recommendation with Topic Models

Our approach focuses on combining structured taxonomies and folksonomies with latent topics; see Figure 3. By enabling periodic processing of information resources and unsupervised generation of latent topics, extracted topics are used as a basis for enhancing search and recommendation functionalities.

3.2.1 Information Search

Searching within an organisational information repository is a demanding application. A major challenge is that the words used in the queries sometimes do not necessarily appear in the content deemed relevant by the user. Proposals to address this problem include semantic search and query expansion techniques which try to make use of synonyms, idiosyncratic terms or semantically similar terms in order to be able to retrieve results that are relevant but do not contain the exact words used in the query [Mangold 2007]. Such approaches require the existence of an explicit knowledge structure to assist in resolving term similarities.

Probabilistic topic models can be used as an unsupervised technique for document categorization, and then as a basis for query expansion. In this approach issues like word relations, synonyms and word ambiguity are addressed by statistically discovering relations between words. There is no need for maintaining an explicit knowledge structure; this is derived implicitly by the topic model. A challenge here is that repeatedly calculating probabilistic relations in every query can be inefficient in terms of processing power.

In our approach similarities derived by the probabilistic topics are stored in a word index in order to improve the speed of the query expansion. Our approach implements a thesaurus-based solution in which words found together in a latent topic are considered similar and are stored together with their similarity measure, after a threshold and pruning process is followed. This way, when the user executes a query including a number of terms, her query is expanded by some of the highly related terms.
To compute each document’s score we use (3) as described in [Park et al. 2009]

$$S_d(Q) = \mu S_d(E) + (1-\mu) S_d(Q)$$

In (3), $Q$ is the query terms and $E$ is the expansion based on the topic model thesaurus. $S_d$ is the document scoring results as received by the underlying social software and $\mu$ ($0<\mu<1$) is the mixing parameter.

### 3.2.2 Information Recommendation

LDA produces two probabilistic distributions: topics over words and documents over topics which are approximated as illustrated above. The documents over topics distributions are used as similarity measures for relevant resource suggestion. The resources that were in the initial training corpus are already described in topics. In the case of a new, previously unseen, resource the model infers which of the latent topics are related to the resource. To describe the resources, the distribution over topics is used in the form of a vector. This vector has as many dimensions as the topics found in the corpus. When a new resource is added to the system, its topic distribution is inferred based on the words observed. To calculate the semantic distance between resources we calculate the cosine similarity between their corresponding topic distribution vectors.

In (4), $\theta_A$ and $\theta_B$ are the topic distribution vectors of the resources $A$ and $B$ and are used to calculate their similarity.

$$\text{Similarity}(A, B) = \cos(\theta_A, \theta_B) = \frac{\theta_A \cdot \theta_B}{\|\theta_A\| \|\theta_B\|}$$

### 3.2.3 Tag Recommendation

Tagging is the assignment of keywords to a resource. Social business software typically allows for free tagging, which leads to the formation of folksonomies. Still, many knowledge management systems make use some kind of taxonomies or catalogues of terms. Processing and analysing folksonomies is a field that has been intensely researched as in [Hotho 2006].

The first tag recommendation technique we use in our approach is based on the similarity measures calculated in resource recommendation. The system infers the related resources from the topic distribution, and then proposes the tags already assigned to highly relevant items. This approach reuses highly used tags (or terms in case of explicit knowledge structures such as taxonomies), therefore reinforcing the model of terms used before in the system.

In (5), the vector containing the relevance of each tag to document $A$ is calculated by multiplying the matrix containing the tagging of each document ($TD$), with the complete document topic distribution in the dataset ($\Theta$), and in turn with the vector describing the topic distribution of the specific document ($A$).

$$\text{tagSim}_A = TD \cdot \Theta \cdot \theta_A$$

The second technique is based on the fact that topics that emerge from the analysis are represented by distributions of words. In the context of tag recommendations, these distributions can become the keyword pools. The dominant words of a topic can be used in order to categorize resources that are in some degree generated by this topic.

This technique can work both with folksonomies and taxonomies. This calculation is expressed in (6) where the vector describing the topic distribution of the specific document ($A$), is multiplied by the word topic distribution matrix in the dataset.

$$\text{tagSim}_A = \theta_A^T \cdot \Phi$$

Moreover, these two techniques can be combined to propose the list of tags using a weighted combination approach [Burke et al. 2002]. This hybrid recommender can both support an existing taxonomy and suggest new keywords based on dominant words in an evolving knowledge base.
4 Case Study

We have tested our approach by integrating it within an open source social business system, the OrganiK system [Christidis et al. 2011]. OrganiK provides a collection of social applications for the corporate environment and supports both taxonomies and folksonomies for annotating information resources, while it builds on and extends the prominent open source content management system Drupal. OrganiK has been deployed in and used by five small and medium enterprises including a translation/localisation services company, two information technology service companies, a content provider and a maritime consulting company. The companies had an active role in the development, deployment and the eventual use of the OrganiK system. Employees were present in the early discussions on how the system would work. Additionally during the development process, the evolving system was available to users, loosely following the perpetual beta paradigm. After the system reached a stable state, the users started using it in their daily work.

![Case Study Image](image-url)

**Figure 4:** Knowledge Structures and Latent Topics

A screenshot from the maritime consulting installation can be found in Figure 4. Assume an employee finds a new resource in the World Wide Web that is of interest to the enterprise. He inserts it together with his personal comments in the enterprise collaborative platform in the text area of a blog (Region 1 of Figure 4). The text content of the resource is analyzed in the background and relations to the existing topics are inferred. The user can then locate related resources, which are suggested based on the latent topics. Suggestions appear in Region 2 of Figure 4 and evolve as the user types in new information. The user can enhance the content of his resource by adding information from other relevant items or linking to them. As the user types in text in the page, tags are suggested (Region 3 of Figure 4). The recommender is assisting the employee to categorize this resource. Tags are derived from both the corporate taxonomy and the latent topic wording. A few minutes later, another user looks for hazard. This word is not contained in the page that the previous user wrote, however using latent topics for query expansion, related articles emerge – including the previous one. The query is expanded to cover related documents by using closely related words based on the latent topics.

5 Evaluation Results

We monitored system usage in all five companies. The system was used by a total of 32 users during a period of six months. Below we present the evaluation of the system in one of the five companies for reasons of brevity.
Users were asked to evaluate the performance of the search and recommendation functionalities as well as of specific social software features. To this end, two types of questionnaires were used: one containing specific questions about the system features to which users provided their responses on a Likert scale; the second contained open ended questions related to the usefulness and effectiveness of the system features.

Search was positively evaluated as the company’s employees were positive about the system's ability to locate the resources they were looking for (74 percent of the respondents answered positively, i.e. agree or strongly agree) and retrieve relevant results (83 percent of the respondents answered positively).

Recommendations were also positively accepted. 78 percent of respondents indicated that the system helped them notice all possible tags for annotating their documents while 73 percent found that the suggested tags were suitable for annotating their documents. With regards to resource recommendations, 78 percent of respondents indicated that the system helped them locate all resources they were looking for and 80 percent found that the suggested resources were relevant to their needs.

6 Related Work

Search and recommender systems in social environments, both inside and outside the enterprise, have recently been a subject of intensive research. [Dmitriev et al. 2006] have suggested the use of implicit and explicit annotations as user feedback for improving the enterprise search, while [Amitay et al. 2009] have proposed ways to combine heterogeneous information in order to augment search functionality. [Dugan et al. 2007] propose a social bookmark recommendation system that takes the form of a game and [Guy et al. 2010] examine the recommendation of social media resources in the enterprise based on people, tags, and their aggregate relationships. [Anderson & Mohan 2011] studied four knowledge-intensive firms and their use of social networking for knowledge management; [Guy et al. 2011] have designed and evaluated a recommendation system for recommending strangers to employees. Furthermore, semantically enhanced ways to search and recommendation have been proposed in the SemSLATES approach of [Passant et al 2009]. These approaches are mostly using heterogeneous user generated content, social graph information and semantic technologies. In this paper we additionally utilise the results of latent topic analysis in order to capture hidden semantics in content.

Probabilistic topic models have already been used as a way to support recommender systems and search. In [Jin et al. 2004], the authors used probabilistic latent semantic analysis for supporting the mining of usage patterns. [Chen et al. 2009] and [Haruechayyasak & Damrongrat 2008] applied topic modes for recommending users within communities and articles respectively.

In [Krestel 2009] and [Diaz-Aviles 2010], the authors propose methodologies for suggesting annotations to users while [Tsai 2011] used latent topics to mine blogs and recommend relevant tags for given blog posts. In the enterprise environment, [Zhao 2010] proposed the use of latent topics in order to identify and visualize latent communities of interest. In [Schirru 2010], an approach of using latent topics has been proposed in order to capture interests of the employers and recommend documents.

To the best of our knowledge, our approach is the first that not only uses a latent topic model for both search and recommendation in an social enterprise environment, but also utilises it in order to support the generation and refinement of knowledge structures.

7 Conclusions and future research

In this paper we exploited knowledge structures in social business software in order to enhance search and recommendation of information resources. We used statistical topic models to
improve query expansion and recommend relevant resources and tags, which in turn can leverage the creation and evolution of knowledge structures including taxonomies and folksonomies. Our approach does not require significant effort from users as documents evolve to cover diversifying subjects and is easily scalable to a large number of documents.

As future work, we intend to use alternative probabilistic topic modelling methods that can provide insight into additional qualities in the document topics, e.g. by capturing correlation between topics [Blei 2005] or topic hierarchies [Blei 2004].

Acknowledgement
This research has been partially financed by the European Commission through the FP7 OrganiK project (222225).

References


