Grisham: Topic-Based Publication Exploration

Authors

Abstract

We describe a demonstration of our system *Grisham*, presenting methods for topic-based paper exploration. We use a popular topic modeling algorithm, Latent Dirichlet Allocation, to derive topic distributions for each scientific paper in a DBLP citation data set. We allow users to specify personal topic distribution to contextualize the exploration experience. We demonstrate three types of exploration *keyword-based search*, *topic-based exploration*, and *citation-lineage search*. In each model we shape the results to be specific to a user specification. We describe the components of our web-based system.

Introduction

In a variety of situations, from literature surveys to legal document collections, people try to organize and explore large amounts of documents. Current technology to search on documents are done based on keywords with minor extensions. Sometimes in order to enable keyword search, external effort—What are they? – CPG— has to be put in to tag the documents with relevant keywords. Keyword-based search is useful when the user knows exactly what he or she is looking for. It is not particularly useful when a user wants to explore or learn a new topic. This is especially important when for example, a researcher wants to find out the state of the art in a particular area or a student would like to create a literature survey. In such situations, *topic-based search* can return more relevant results. Topic-based search is a classification of the search space to highlight topical relevance. In order to accomplish this, the topics underlying the document collection need to be extracted, and then they have to be represented in terms of those topics and ranked based on relevance to a particular topic.

In our system called *Grisham* we present various techniques for topic-based exploration and search of peer reviewed scientific papers. Our work allows user to search for scientific papers using three methods: First, users may perform a traditional search *keyword-based search* for papers. Second, users can specify a topic or topics he or she is interested in and the most relevant papers for that topic will be listed in the order of their relevance by our system. Lastly, users are allowed to explore similar or related documents using a visual graph like interface, i.e., given a paper or an article, a user will be shown a set of papers which cite the original paper in the order of their relevance to the topic. In addition, we allow the user to specify a topic distribution that specifies their interest. In all three methods, we adapt search results to personalize results.

The main contributions of our work are the *user-topic ranking function* which ranks the relevance of documents to a set of topics—I think we need to edit this; a similar ranking function is there in George et al. 2012 – CPG, the *similar document explorations* which is performed by computing the similarity of papers to a topic of interest, and *topic-document visualization* which ranks citations for a paper by the interest of the user.

Copyright © 2015, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Related Work

There have been previous systems to use topic modeling as a basis of search and exploration. We are the first to our knowledge to incorporate a user modeling into the loop.

A system of note is Yang et al. (Yang, Torget, and Mihalcea 2011) whom apply topic modeling to collections of historical news papers to assist search. They found that the topics generated from topic models are are generally good, however once the sets of topics are generated, an expert opinion is required to name them. In *Grisham*, we allow users to select numbered topics for article-search based on topically relevant words.

Topic Models

Topic modeling allows us to represent the properties of a large collection of documents containing numerous words with a small collection of topics. A topic is represented by a distribution of words in a vocabulary. This approach to topic modeling hinges on the bag-of-words model in which word occurrences within a document are assumed to be independent of each other. Latent Dirichlet Allocation (LDA) is a well known, probabilistic topic model, which can represent hidden topic structues of the documents in a text corpus (Blei, Ng, and Jordan 2003). Due to its fully generative semantics, even at the level of documents, LDA could address the drawbacks of other topic models such as the Latent Semantic Analysis (LSA) and Probabilistic LSA (Hofmann

1999; Blei, Ng, and Jordan 2003). In LDA, each document is described by a mixture of topics, and words are chosen from the multinomial that results from the mixture of the documents topic multinomials. The topics themselves, as well as the documents are drawn from Dirichlet distributions.

Topic model analysis depends upon exploring the posterior distribution of model parameters and hidden variables conditioned on observed words. The model parameters are corpus-level topics and document-level topic mixtures. The topic models such as LSA and LDA assumes that we know a-priori the number of topics in the corpus. People have used different approximate inference methods such as deterministic approaches – replace the posterior integral with a tractable lower bound (Blei, Ng, and Jordan 2003), and sampling methods – approximate the posterior integral using an empirical average (Griffiths and Steyvers 2004), for the inference with LDA.

Grisham

This section will be split into **SEARCH** and **EXPLORATION** – CEG

Grisham is a web-based system build on top of the PostgreSQL open source database system. Computation for visualization is performed using HTML, CSS and javascript. Calculations are performed using in-database computation through AJAX calls and client side javascript. In this section we discuss each part of *Grisham*.

Data pre-processing and topic learning

First, we tokenize the raw text based on the python NLTK toolkit and a predefined regular expression. Second, we standardize them by removing noise terms and stop-words. Third, we represent each document in a sparse bag-of-words format, after building a vocabulary of corpus-words. Last, we use them as input to the topic learning model which will in turn learn clusters from the term co-occurrence frequencies of the corresponding documents. In this project, we used the Gensim package implementation of the LDA online learning algorithm (Řehůřek and Sojka 2010; Hoffman, Blei, and Bach 2010) for topic learning, which is based on the variational inference framework.

Components of a learned topic model includes the corpus-level topic word association counts and document-level topic mixtures. Each estimated topic is represented by its topic-word-counts, which is useful for our automatic detection of paper topics. The document-level topic mixtures give an idea of the topicality of a particular paper given a topic. This is also quite useful in finding similar papers and grouping them together, because topic modeling is a type of dimensionality reduction technique that enables us to work on the topic-space rather than on the vocabulary-space.

User Model

When performing search, exploration and discovery over academic papers users may bring particular context to their search. Incorporating this information into the search process has been show to be beneficial to users (Dou, Zhicheng and Song, Ruihua and Wen, Ji-Rong 2007; Ma, Zhongming and Pant, Gautam and Sheng, Olivia R. Liu 2007). We develop a user model that encapsulates the users personal context and integrates it into their search task.

This model is a distribution of weights for each topic. Formally, given a set of topics T the user model is defined as

$$\mathcal{U} = \{u_0, \ldots, u_{|T|}\}$$

where $u_i \in [0, 1]$ and $\sum_{t \in T} u_t = 1$. We graphically allow the user to select the weights that correspond to each topic. This allows the users to change preferences with each query for more desirable results.

The user model is used in different ways to provide better feedback to the user. After a keyword search, the document results of the search are re-ranked by calculating the KL-divergence of each document and the user model. Formally, given the set of result documents D:

$$KL(\mathcal{U}||d) = \sum_{t \in T} u_t \ln \frac{u_t}{d_t}. \quad (1)$$

where $d \in D$ and d_t is the topic proportion for document d and topic $t \in T$.

In the topic explorer, each topic row is color-coded like a heat based map based on the similarity of the user model to that topic (see Figure 1). The user can look at this heat map to adjust their topic preferences. We use equation 1 on the client side to calculate this preference. In the graph explorer the citations for the current paper is ranked using equation 1. The citations of that paper that are most similar to the user model are ranked the highest.

Ranking Function

We provided several ranking functions to let the user find the best papers.

One way of ranking is to determine relevant papers given an estimated topic. We use individual papers' document topic mixtures, $\hat{\theta}_d$, to rank them on relevance given a topic. For a given topic $t \in K$, we calculate

$$m(d) = \ln \hat{\theta}_{d,t} + \sum_{j \neq t} \ln(1 - \hat{\theta}_{d,j}) \quad (2)$$

for all documents $d = 1, 2, \ldots, D$ in the document collection, and sort them to rank them on relevance. Here, we assume that each $\hat{\theta}_d$ is normalized, i.e., $\sum_{j=1}^{K} \hat{\theta}_j = 1$. Intuitively, we can see that this equation maximizes the probability of a topic $t \in K$ given a document. That means a document with a higher value of this score is highly relevant for the selected topic t, and contains a considerable amount of words from the topic distribution.

Demonstration

Note this section will be spread out into the other sections. – CEG

Grisham is demonstrated with the help of a web page which allows the user to perform exploratory search on DBLP conference's scientific papers (Tang et al. 2008). All the searches in the system keep in account the user model. The user model is a profile of preference provided by the

user before he makes any search. This is done by specifying weightage to various topics. The *Grisham* website has a list of topics with a slider associated with each of them using which a user can specify the degree of interest in that particular topic. This array of user preference is used as the ranking factor in all the results. The web site has three basic functionality which has been classified under three different tabs of the same name. They are keyword paper search, Topic explore, and Graph explore.

Keyword Paper Explore This is a universal search facility. The user may enter one or more keywords representing topics, or author names etc. The key words are searched in the title, abstract, and the author names of all the papers and are listed. The listing is ranked based on the user model. In the first page only a few of the papers are displayed in two boxes - one containing matching words in the title, and the other in the abstract. Clicking on any box will open up a more complete list of papers.

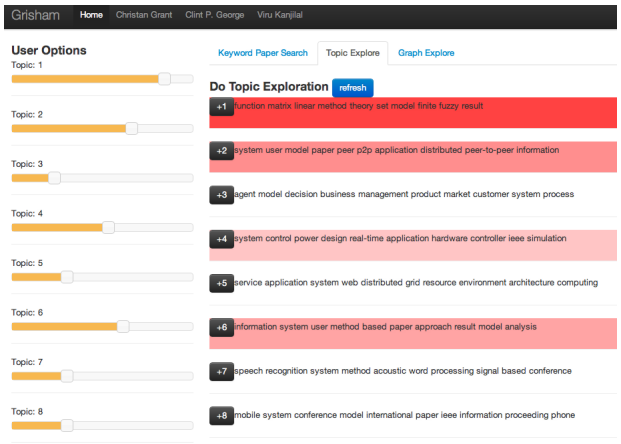

Figure 1: User configured topic exploration

Topic Explore The second tab on the website is for topic exploration and it allows a user to click on a specific topic to know more about the papers associated with the topic. Initially all the extracted topics from the corpus are shown along with their topic words. This list is color coded to distinguish the relevance of topics as indicated by the user model. The list is clickable and one a topic is clicked, relevant papers to that topic ranked based on the user model and displayed. A screenshot is shown in figure 1.

Graph Explore One of the most interesting visualizations in the project is the graph explore which allows a user to conceptually drill down a graph of a paper and its citations in a recursive sequence. This is a common method of literature exploration where a user takes up a base paper and then reads up all the papers which have been cited in that base paper. This is recursively performed on the secondary papers also. Though effective, this technique tell the user which ones he should pursue and which ones he should now. The graph explore functionality allows a user to perform this in a more visually appealing and relevant manner. Once a user decides on a base paper and enters it in the system, the system will show a graph representation of its citations which are ranked based on his profile (user model). This will help him pursue the most relevant papers first. Clicking on any secondary paper will open up the graph further and list it's citations ranked taking into account the user model.

Discussion

In this section we discuss the pros and cons of the two methods.

Conclusion

We describe *Grisham*, a system for topic-based paper search and exploration given a user model. This is a demonstration of a promising new search paradigm. Any research who would like to do exploratory search or a literature reviews will find the system beneficial. During the demonstration we will allow participants to freely interact with the system. We will discuss the algorithm and formulas that drive *Grisham* with attendees.

References

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. Journal of Machine Learning Research 3:993–1022.

Dou, Zhicheng and Song, Ruihua and Wen, Ji-Rong. 2007. A large-scale Evaluation and Analysis of Personalized Search Strategies. In Proceedings of the 16th international conference on World Wide Web, WWW '07, 581–590. New York, NY, USA: ACM.

George, C.; Wang, D.; Wilson, J.; Epstein, L.; Garland, P.; and Suh, A. 2012. A machine learning based topic exploration and categorization on surveys. In Machine Learning and Applications (ICMLA), 2012 11th International Conference on, volume 2, 7–12.

Griffiths, T. L., and Steyvers, M. 2004. Finding scientific topics. Proceedings of the National Academy of Sciences 101:5228–5235.

Hoffman, M.; Blei, D. M.; and Bach, F. 2010. Online learning for latent dirichlet allocation. In NIPS.

Hofmann, T. 1999. Probabilistic latent semantic indexing. In SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 50–57. New York, NY, USA: ACM.

Ma, Zhongming and Pant, Gautam and Sheng, Olivia R. Liu. 2007. Interest-based Personalized Search. ACM Trans. Inf. Syst. 25(1).

Řehůřek, R., and Sojka, P. 2010. Software Framework for Topic Modelling with Large Corpora. In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, 45–50. Valletta, Malta: ELRA.

Tang, J.; Zhang, J.; Yao, L.; and Li, J. 2008. Extraction and mining of an academic social network. In Proceedings of the

17th international conference on World Wide Web, WWW '08, 1193–1194. New York, NY, USA: ACM.

Yang, T.; Torget, A.; and Mihalcea, R. 2011. Topic modeling on historical newspapers. ACL HLT 2011 96.