Int J Digit Libr (2015) 16:111–128
DOI 10.1007/s00799-015-0146-2

A generalized topic modeling approach for automatic document annotation

Suppawong Tuarob · Line C. Pouchard ·
Prasenjit Mitra · C. Lee Giles

Received: 31 October 2013 / Revised: 16 February 2015 / Accepted: 25 February 2015 / Published online: 7 March 2015

Abstract Ecological and environmental sciences have become more advanced and complex, requiring observational and experimental data from multiple places, times, and thematic scales to verify their hypotheses. Over time, such data have not only increased in amount, but also in diversity and heterogeneity of the data sources that spread throughout the world. This heterogeneity poses a huge challenge for scientists who have to manually search for desired data. ONEMercury has recently been implemented as part of the DataONE project to alleviate such problems and to serve as a portal for accessing environmental and observational data across the globe. ONEMercury harvests metadata records from multiple archives and repositories, and makes them searchable. However, harvested metadata records sometimes are poorly annotated or lacking meaningful keywords, which could impede effective retrieval. We propose a methodology that learns the annotation from well-annotated collections of metadata records to automatically annotate poorly annotated ones. The problem is first transformed into the tag recommendation problem with a controlled tag library. Then, two variants of an algorithm for automatic tag recommendation are presented. The experiments on four datasets of environmental science metadata records show that our methods perform well and also shed light on the natures of different datasets. We also discuss relevant topics such as using topical coherence to fine-tune parameters and experiments on cross-archive annotation.

Keywords Metadata annotation · Topic model · Tag recommendation

1 Introduction

Environmental sciences have become both complex and data intensive, needing access to heterogeneous data collected from multiple places, times and thematic scales. For example, research on bird migration would involve exploring and analyzing observational data such as the migration of animals and temperature shifts across the world, from time to time. While the needs to access such heterogeneous data are apparent, the rapid expansion of observational data, in both quantity and heterogeneity, poses huge challenges for data seekers to obtain the right information for their research. Such problems behoove tools that automatically manage, discover, and link big data from diverse sources, and present the data in forms that are easily accessible and comprehensible.

1.1 ONEMercury search service

Recently, DataONE, a federated data network built to facilitate access to and preserve environmental and ecological

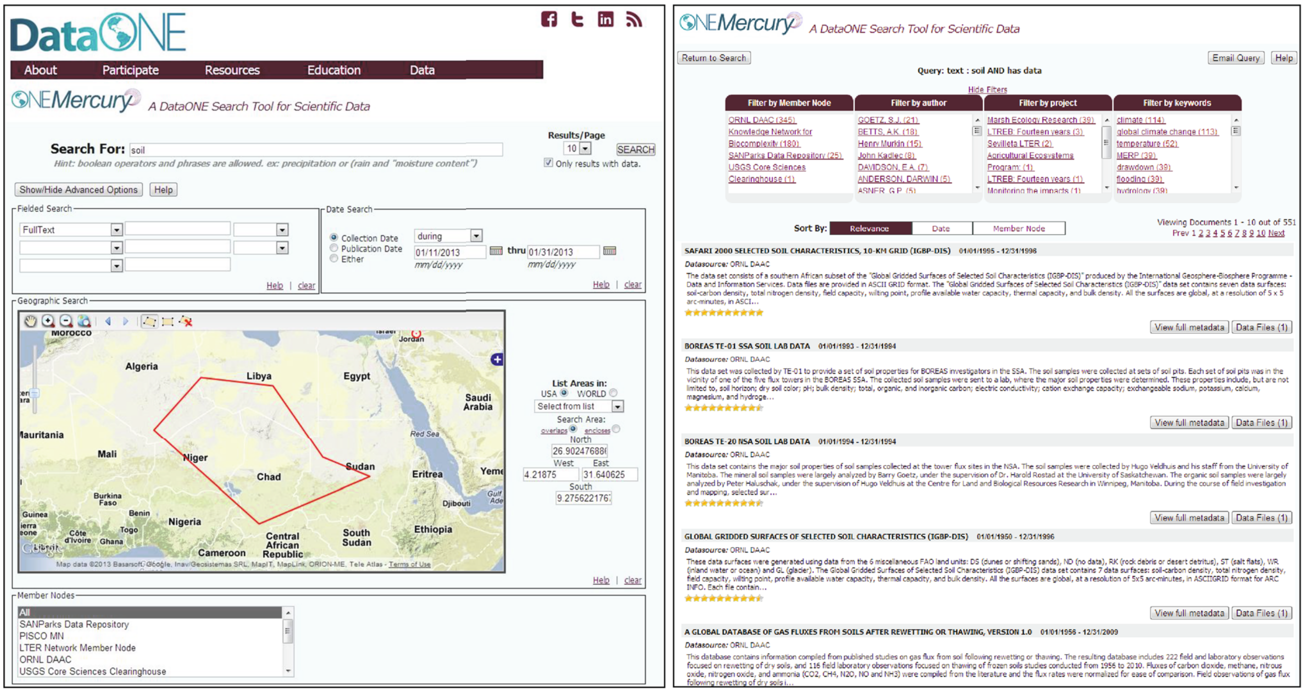
This manuscript is an extension of the authors' earlier work presented at the 13th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2013) [25].

S. Tuarob (✉) · C. L. Giles
Computer Science and Engineering, The Pennsylvania State University, University Park, PA, USA
e-mail: suppawong@psu.edu

C. L. Giles
Information Science and Technology, The Pennsylvania State University, University Park, PA, USA
e-mail: giles@ist.psu.edu

L. C. Pouchard
Purdue University, West Lafayette, IN, USA
e-mail: pouchard@purdue.edu

P. Mitra
Qatar Computing Research Institute, Doha, Qarta
e-mail: pmitra@qf.org.qa



Search For: soil

Hint: boolean operators and phrases are allowed. ex: precipitation or (rain and "moisture content")

Results/Page: 10

☑ Only results with data.

Fielded Search: FullText

Date Search: ● Collection Date ○ Publication Date ○ Either — during 01/11/2013 thru 01/31/2013 (mm/dd/yyyy)

List Areas in: USA ● WORLD ○ — Select from list

Search Area: overlaps ● encloses ○

North: 26.90247688(West: 4.21875 East: 31.640625 South: 9.27562217б

Member Nodes: All; SANParks Data Repository; PISCO MN; LTER Network Member Node; ORNL DAAC; USGS Core Sciences Clearinghouse

Query: text : soil AND has data

Filter by Member Node	Filter by author	Filter by project	Filter by keywords
ORNL DAAC (345)	GOETZ, S.J. (21)	Marsh Ecology Research (39)	climate (114)
Knowledge Network for Biocomplexity (180)	BETTS, A.K. (18)	LTREB: Fourteen years (3)	global climate change (113)
SANParks Data Repository (25)	Henry Murkin (15)	Sevilleta LTER (2)	temperature (52)
USGS Core Sciences Clearinghouse (1)	John Kadlec (8)	Agricultural Ecosystems Program: (1)	MERP (39)
	DAVIDSON, E.A. (7)	LTREB: Fourteen years (1)	drawdown (39)
	ANDERSON, DARWIN (5)	Monitoring the impacts (1)	flooding (39)
	ASNER, G.P. (5)		hydrology (39)

Viewing Documents 1 - 10 out of 551
Prev 1 2 3 4 5 6 7 8 9 10 Next

SAFARI 2000 SELECTED SOIL CHARACTERISTICS, 10-KM GRID (IGBP-DIS) 01/01/1995 - 12/31/1996

Datasource: ORNL DAAC

The data set consists of a southern African subset of the "Global Gridded Surfaces of Selected Soil Characteristics (IGBP-DIS)" produced by the International Geosphere-Biosphere Programme - Data and Information Services. Data files are provided in ASCII GRID format. The "Global Gridded Surfaces of Selected Soil Characteristics (IGBP-DIS)" data set contains seven data surfaces: soil-carbon density, total nitrogen density, field capacity, wilting point, profile available water capacity, thermal capacity, and bulk density. All the surfaces are global, at a resolution of 5 x 5 arc-minutes, in ASCI...

BOREAS TE-01 SSA SOIL LAB DATA 01/01/1993 - 12/31/1994

Datasource: ORNL DAAC

This data set was collected by TE-01 to provide a set of soil properties for BOREAS investigators in the SSA. The soil samples were collected at sets of soil pits. Each set of soil pits was in the vicinity of one of the five flux towers in the BOREAS SSA. The collected soil samples were sent to a lab, where the major soil properties were determined. These properties include, but are not limited to, soil horizon; dry soil color; pH; bulk density; total, organic, and inorganic carbon; electric conductivity; cation exchange capacity; exchangeable sodium, potassium, calcium, magnesium, and hydroge...

BOREAS TE-20 NSA SOIL LAB DATA 01/01/1994 - 12/31/1994

Datasource: ORNL DAAC

This data set contains the major soil properties of soil samples collected at the tower flux sites in the NSA. The soil samples were collected by Hugo Veldhuis and his staff from the University of Manitoba. The mineral soil samples were largely analyzed by Barry Goetz, under the supervision of Dr. Harold Rostad at the University of Saskatchewan. The organic soil samples were largely analyzed by Peter Haluschak, under the supervision of Hugo Veldhuis at the Centre for Land and Biological Resources Research in Winnipeg, Manitoba. During the course of field investigation and mapping, selected sur...

GLOBAL GRIDDED SURFACES OF SELECTED SOIL CHARACTERISTICS (IGBP-DIS) 01/01/1950 - 12/31/1996

Datasource: ORNL DAAC

These data surfaces were generated using data from the 6 miscellaneous FAO land units: DS (dunes or shifting sands), ND (no data), RK (rock debris or desert detritus), ST (salt flats), WR (inland water or ocean) and GL (glacier). The Global Gridded Surfaces of Selected Soil Characteristics (IGBP-DIS) data set contains 7 data surfaces: soil-carbon density, total nitrogen density, field capacity, wilting point, profile available water capacity, thermal capacity, and bulk density. All the surfaces are global, at a resolution of 5x5 arc-minutes, in ASCIIGRID format for ARC INFO. Each file contain...

A GLOBAL DATABASE OF GAS FLUXES FROM SOILS AFTER REWETTING OR THAWING, VERSION 1.0 01/01/1956 - 12/31/2009

Datasource: ORNL DAAC

This database contains information compiled from published studies on gas flux from soil following rewetting or thawing. The resulting database includes 222 field and laboratory observations focused on rewetting of dry soils, and 116 field laboratory observations focused on thawing of frozen soils studies conducted from 1956 to 2010. Fluxes of carbon dioxide, methane, nitrous oxide, nitrogen oxide, and ammonia (CO2, CH4, N2O, NO and NH3) were compiled from the literature and the flux rates were normalized for ease of comparison. Field observations of gas flux following rewetting of dry soils i...

Fig. 1 Screen shots of the ONEMercury search interface and result page using the query ‘soil’

science data across the world, has become increasingly popular [18,26,27]. DataONE harvests metadata from different environmental data providers and makes it searchable via the search interface ONEMercury,[1] built on Mercury,[2] a distributed metadata management system. Figure 1 shows sample screen shots of the ONEMercury search interface (left) and the search result page with the search query ‘soil’. ONEMercury offers a full-text search on the metadata records. The user can also specify the boundary of locations in which the desired data are collected or published using the interactive graphic map. At the result page, the user can choose to further filter out the results by Member Node, Author, Project, and Keywords. The set of keywords used in the system is static (users cannot arbitrarily add new or remove the existing keywords) and managed by the administrator to prevent spurious, new keywords from being created. Such keywords are used for manually annotating metadata during the data curation process.

1.2 Challenge and proposed solution

Linking data from heterogeneous sources always have a cost. One of the biggest problems that ONEMercury is facing is the different levels of annotation in the harvested metadata records caused by different metadata curation standards. For example, a data center may have specialized personnel whose sole duty is to provide rich description and useful keywords for each metadata record, while another data center collects data directly from scientists who are busy with their experiments and do not have time to curate their data. Poorly annotated metadata records tend to be missed during the search process as they lack meaningful keywords. Furthermore, such records would not be compatible with the advanced mode offered by ONEMercury as it requires the metadata records to be annotated with predefined keywords from the keyword library. The explosion of the amount of metadata records harvested from an increasing number of data repositories makes it impossible to annotate them manually by hand, necessitating the need for a tool capable of automatically annotating these poorly annotated metadata records.

In this paper, we address the problem of automatic annotation of metadata records. Our goal is to build a fast and robust system that annotates a given metadata record with related keywords from a given keyword library. The idea is to annotate a given record with keywords associated to the well-annotated records that it is semantically relevant to. We propose a solution to this problem by first transforming the problem into the tag recommendation problem with a controlled tag library, where the set of recommended tags is used to annotate the given document, and then propose a set of algorithms that deal with the problem.

1.3 Problem definition

We define a document as a tuple of textual contents and a set of tags. That is $\mathrm{d} = \langle \mathrm{c}, \mathrm{e} \rangle$, where c is the textual content,

[1] https://cn.dataone.org/onemercury/.

[2] http://mercury.ornl.gov/.

represented by a sequence of terms, of the document d and e is a set of tags associated with the document. Given a tag library T, a set of annotated documents D, and a non-annotated query document q, our task is to recommend a ranked list of K tags taken from T to the query q. A document is said to be annotated if it has at least one tag; otherwise, it is non-annotated. The formal description of each variable is given below:

$$T = \{t_1, t_2, \ldots, t_M\}\,;\, t_i \text{ is a tag.}$$
$$D = \{d_1, d_2, \ldots, d_N\}\,;\, d_i = \langle c_{di}, e_{di}\rangle\,,\, e_{di} \subseteq T,\ \text{and } e_{di} \neq \oslash$$
$$q = \langle c_q, \oslash\rangle$$

1.4 Contributions

This paper has five key contributions as follows:

1. We address a real-world problem of metadata annotation faced by ONEMercury. We transform the problem into the tag recommendation problem and generalize the problem so that the proposed solution can further be applied to other domains.
2. We propose a novel technique for tag recommendation. Given a document query q, we first compute the distribution of tags. The top tags are then recommended. We propose two variants of our algorithms: term frequency-inverse document frequency (TF-IDF) based and topic model (TM) based.
3. We crawl environmental science metadata records from four different archives for our datasets: the Oak Ridge National Laboratory Distributed Active Archive Center (DAAC),[3] Dryad Digital Repository,[4] the Knowledge Network for Biocomplexity (KNB),[5] and TreeBASE: a repository of phylogenetic information.[6] We select roughly 1000 records from each archive for the experiments.
4. We validate the proposed methodology using rigorous empirical evaluations. We use document-wise tenfold cross-validation to evaluate our methods with five evaluation metrics: precision, recall, F1, MRR (mean reciprocal rank), and BPref (binary preference). These evaluation metrics are typically used together to evaluate recommendation systems.
5. We further discuss relevant issues namely (i) limitations and scalability of our proposed methods, and (ii) using topical coherence to fine-tune the optimal parameters.

2 Preliminaries

Our proposed solution is built upon the concepts of Cosine Similarity, term frequency-inverse document frequency (TF-IDF), and latent Dirichlet allocation (LDA). We briefly introduce them here before going further.

2.1 Cosine similarity

Cosine similarity is a measure of similarity between two vectors obtained by measuring the cosine of the angle between them. Given two vectors A and B, the cosine similarity is defined using a dot product and magnitude as:

$$\begin{aligned}\text{CosineSim}(A, B) &= \frac{A \cdot B}{\|A\|\,\|B\|} \\ &= \frac{\sum_{i=1}^{N} A_i \times B_i}{\sqrt{\sum_{i=1}^{N}(A_i)^2} \times \sqrt{\sum_{i=1}^{N}(B_i)^2}} \end{aligned} \tag{1}$$

In information retrieval literature [16], the cosine similarity is heavily used to calculate the similarity between two vectorized documents. An assumption is made that each element in a document vector is a real non-negative number (such as term frequency and TF-IDF score), hence `CosineSim(A,B)` outputs `[0,1]`, with the value indicating the level of similarity.

2.2 Term frequency-inverse document frequency

TF-IDF, used extensively in the information retrieval field [16,29], quantifies how important a term is to a document in a corpus. TF-IDF has two components: the term frequency (TF) and the inverse document frequency (IDF). The TF is the frequency of a term appearing in a document. The IDF of a term measures how important the term is to the corpus, and is inversely proportional to the document frequency (the number of documents in which the term appears). Formally, given a term t, a document d, and a corpus (document collection) D:

$$tf(t, d) = \sqrt{\text{count}(t, d)} \tag{2}$$

$$idf(t, D) = \sqrt{\log\left(\frac{|D|}{|d \in D; t \in d|}\right)} \tag{3}$$

$$\text{TFIDF}_{\text{Term}}(t, d, D) = TF(t, d) \cdot \text{IDF}(t, D) \tag{4}$$

We can then construct a TF-IDF vector for a document d given a corpus D as follows:

$$\text{TFIDF}_{\text{Doc}}(d, D) = \langle \text{TFIDF}_{\text{Term}}(t_1, d, D), \ldots, \text{TFIDF}_{\text{Term}}(t_n, d, D)\rangle \tag{5}$$

Consequently, if one wishes to compute the similarity score between two documents d_1 and d_2, the cosine similarity can

[3] http://daac.ornl.gov/.

[4] http://datadryad.org/.

[5] http://knb.ecoinformatics.org/index.jsp.

[6] http://treebase.org/treebase-web/home.html.

be computed between the TF-IDF vectors representing the two documents:

$$\begin{aligned}&\mathrm{DocSim}_{\mathrm{TF-IDF}}(d_1, d_2, D)\\&\quad = \mathrm{CosineSim}\left(\mathrm{TFIDF}_{\mathrm{Doc}}(d_1, D), \mathrm{TFIDF}_{\mathrm{Doc}}(d_2, D)\right)\end{aligned} \quad (6)$$

2.3 Latent Dirichlet allocation

In text mining, latent Dirichlet allocation (LDA) [3] is a generative model that allows a document to be represented by a mixture of topics. Past literature [12,28,30–33] demonstrates successful usage of LDA to model topics from given corpora. The basic intuition of LDA is that an author has a set of topics in mind when writing a document. A topic is defined as a distribution of terms. The author then chooses a set of terms from the topics to compose the document. The whole document can then be represented using a mixture of different topics. LDA serves as a means to trace back the latent topics in the author's mind before the document is written. Mathematically, the LDA model is described as follows:

$$P(t_i|d) = \sum_{j=1}^{|Z|} P(t_i|z_i = j) \cdot P(z_i = j|d) \quad (7)$$

$P(t_i|d)$ is the probability of term t_i being in document d. z_i is the latent (hidden) topic. $|Z|$ is the number of all topics. This number needs to be predefined. $P(t_i|z_i = j)$ is the probability of term t_i being in topic j. $P(z_i = j|d)$ is the probability of picking a term from topic j in the document d.

Essentially, the LDA model is used to find $P(z|d)$, the topic distribution of document d, with each topic being described by the distribution of term $P(T|z)$. After the topics are modeled, we can assign a distribution of topics to a given document using *statistical inference* [2]. A document then can be represented with a vector of numbers, each of which represents the probability of the document belonging to a topic.

$$\mathrm{Infer}(d, Z) = \langle z_1, z_2, \ldots, z_Q \rangle; \quad |Z| = Q \quad (8)$$

where Z is a set of topics, d is a document, and z_i is a probability of the document d falling into topic i. Since a document can be represented using a vector of real non-negative numbers, one can then compute the topic similarity between two documents d_1 and d_2 using cosine similarity as follows:

$$\begin{aligned}&\mathrm{DocSim}_{\mathrm{TM}}(d_1, d_2, Z)\\&\quad = \mathrm{CosineSim}\left(\mathrm{Infer}(d_1, Z), \mathrm{Infer}(d_2, Z)\right)\end{aligned} \quad (9)$$

3 Related works

The literature on document annotation is extensive. Hence we only present the work closely related to ours.

3.1 Automatic document annotation

Newman et al. [20] discuss approaches for enriching metadata records using probabilistic topic modeling. Their approach treats each metadata record as a bag of words and consists of two main steps: (1) generate topics based on a given corpus of metadata, and (2) assign relevant topics to each metadata record. Hence, a metadata record is annotated by the top terms representing the assigned topics. They propose three variations of their approaches. The first method, which they use as the baseline, uses full vocabulary (every word) from the corpus. The remaining two methods filter out the vocabulary by deleting useless words resulting in more meaningful topics. They compare the three approaches in three aspects: % of usable topics, % enhanced records, and average coverage by the top 4 chosen topics. They acquire the datasets from 700 repositories, hosted by OAISter Digital Library. The results show that, overall, the second method performs the best. However, such methods require manual modification of the vocabulary, hence would not scale well. The third method performs somewhere in between.

Bron et al. [4] address the problem of document annotation by linking a poorly annotated document to well-annotated documents using TF-IDF cosine similarity. One corpus consists of textually rich documents (A_s) while the other contains sparse documents (A_t). In the paper, they address two research problems: document expansion and term selection. For the document expansion task, each targeted document (a document in the sparse set) is mapped to one or more documents in the rich set, using simple cosine-similarity measure. Top N documents are chosen from the rich corpus, and the texts in these documents are added to the targeted documents as supplemental content. The term selection task was introduced because using the whole documents from the source corpus to enrich the targeted document might be too spurious and have a fair chance of topic drifts. This term selection task aims to select only meaningful words from each document in the source corpus to add to the targeted documents. Basically, top K % of the words in each document, ranked by TF-IDF scores, are selected as representative words of the document.

This work has a similar problem setting to ours, except that we aim to annotate a query document with keywords taken from the library, while their approaches extract keywords from the full content of documents.

Witten et al. [36] propose *KEA*, a machine learning-based keyphrase extraction algorithm from documents. The algorithm can also be applied to annotate documents with relevant keyphrases. Their algorithm first selects candidate keyphrases from the document. Two features are extracted from each candidate keyphrase: TF-IDF score and distance of the first occurrence of the keyphrase from the beginning of the document. A binary NaiveBayes classifier is trained with the extracted features to build a classification model, which is used for identifying important keyphrases. The algorithm is later enhanced by Medelyan et al. [17] to improve the performance and add more functionality such as document annotation and keyphrase recommendation from control vocabulary, where the list of keyphrases to be recommend is already defined in the vocabulary. In our research, we use keyphrase recommendation with control vocabulary feature of this improved version of the *KEA* algorithm as our baseline.

3.2 Automatic tag recommendation

Since we transform the metadata annotation problem into a tag recommendation problem, we briefly cover related literature. Tag recommendation has gained substantial amount of interest in recent years. Most work, however, focuses on personalized tag recommendation, suggesting tags to a user's object based on the user's preference and social connection. Mishne et al. [19] employ the social connection of the users to recommend tags for weblogs, based on similar weblogs tagged by the same users. Wu et al. [37] utilize the social network and the similarity between the contents of objects to learn a model for recommending tags. Their system aims towards recommending tags for Flickr photo objects. While such personalized schemes have been proven to be useful, some domains of data have limited information about authors (users) and their social connections. Liu et al. [14] propose a tag recommendation model using machine translation. Their algorithm trains the translation model to translate the textual description of a document in the training set into its tags. Krestel et al. [13] employ topic modeling for recommending tags. They use the Latent Dirichlet Allocation algorithm to mine topics in the training corpus where tags are used as the textual content. They evaluate their method against the association rule-based method proposed in [8]. Their method, however, is designed for tag recommendation for social documents where the network of users is assumed to exist, while our methods do not rely on such an assumption.

4 Datasets

We obtain four different datasets of environmental metadata records for the experiments: the Oak Ridge National Laboratory Distributed Active Archive Center (DAAC),[7] Dryad Digital Repository (DRYAD),[8] the Knowledge Network for Biocomplexity (KNB),[9] and TreeBASE: a repository of phylogenetic information (TreeBASE).[10] The statistics of the datasets including the number of documents, total number of tags, average number of tags per document, number of unique tags (tag library size), tag utilization, number of all words (dataset size), and average number of word per document, are summarized in Table 1. Tag utilization is the average number of documents where a tag appears in, and is defined as $\frac{\text{\# all tags}}{\text{\# unique tags}}$. The tag utilization quantifies how often, on average, a tag is used for annotation.

The Oak Ridge National Laboratory Distributed Active Archive Center (ORNL DAAC) is one of the NASA Earth Observing System Data and Information System (EOSDIS) data centers managed by the Earth Science Data and Information System (ESDIS)[11] Project, which is responsible for providing scientific and other users access to data from NASA's Earth Science Missions. The biogeochemical and ecological data provided by ORNL DAAC can be categorized into four groups: Field Campaigns, Land Validation, Regional and Global Data, and Model Archive. After raw data are collected, the data collector describes the data and annotates it using topic-represented keywords from the topic library.

Dryad is a nonprofit organization and an international repository of data underlying scientific and medical publications. The scientific, educational, and charitable mission of Dryad is to promote the availability of data underlying findings in the scientific literature for research and educational reuse. As of January 24, 2013, Dryad hosts 2570 data packages and 7012 data files, associated with articles in 186 journals. Metadata associated with each data package are annotated by the author with arbitrary choices of keywords.

The Knowledge Network for Biocomplexity (KNB) is a national network intended to facilitate ecological and environmental research on biocomplexity. For scientists, the KNB is an efficient way to discover, access, interpret, integrate and analyze complex ecological data from a highly distributed set of field stations, laboratories, research sites, and individual researchers. Each data package hosted by KNB is described and annotated with keywords from the taxonomy by the data collector.

TreeBASE is a repository of phylogenetic information, specifically user-submitted phylogenetic trees and the data used to generate them. TreeBASE accepts all types of phylogenetic data (e.g., trees of species, trees of populations, trees of genes) representing all biotic taxa. Data in TreeBASE are exposed to the public if they are used in a publication that is in press or published in a peer-reviewed scientific journal, book, conference proceedings, or thesis. TreeBASE is produced and governed by the Phyloinformatics Research Foundation, Inc.[12]

[7] http://daac.ornl.gov/.

[8] http://datadryad.org/.

[9] http://knb.ecoinformatics.org/index.jsp.

[10] http://treebase.org/treebase-web/home.html.

[11] http://earthdata.nasa.gov/esdis.

[12] http://www.phylofoundation.org/.

Table 1 Statistics of the 4 datasets

	# Docs	# All Tags	Avg Tags/Doc	# Uniq. Tags	Tag Util.	# All Words	Avg Words/Doc
DAAC	978	7294	7.46	611	11.937	101,968	104.261
DRYAD	1729	8266	4.78	3122	2.647	224,643	129.926
KNB	24,249	254,525	10.49	7375	34.511	1,535,560	63.324
TreeBASE	2635	1838	0.697	1321	1.391	30,054	11.405

In our setting, we assume that the documents are independently annotated, so that the tags in our training sets represent the gold-standard. However, some metadata records may not be independent since they may be originated from the same projects or authors, hence annotated with similar styles and sets of keywords. To mitigate such problem, we randomly select a subset of 1000 annotated documents (except DAAC dataset, which only has 978 documents of land terrestrial ecology, hence we select them all) from each archive for our experiments. We combine all the textual attributes (i.e. `Title`, `Abstract`, `Description`) together as the textual content for the document. We preprocess the textual content in each document by removing 664 common stop words and punctuation, and stemming the words using the Porter2 stemming algorithm.[13]

5 Methodology

The metadata annotation problem is transformed into the tag recommendation problem with a controlled tag library. A document is a tuple of textual information and a set of tags, i.e. ⟨text, tags⟩. A document query is a document without tags, ⟨text, ∅⟩. Specifically, given a tag library $T = \langle t_1, t_2, \ldots, t_m \rangle$, a document corpus $D = \langle d_1, d_2, \ldots, d_n \rangle$, and a document query q, the algorithm outputs a ranked list $T_K^* = \langle t_1, t_2, \ldots, t_K \rangle$, where $t_i \in T$, of K tags relevant to the document query q.

Our proposed algorithm comprises two main steps:

STEP 1 $P(t|q, T, D, M)$, the probability of tag t being relevant to q, is computed for each $t \in T$. M is the document similarity measure, which can be either *TF-IDF* or *TM*.

STEP 2 Return top K tags ranked by the $P(t|q, T, D, M)$ probability.

$P(t|q, T, D, M)$ is the normalization of the relevance score of the tag t to the document query q and is defined as:

$$P(t|q, T, D, M) = \frac{\text{TagScore}_M(t, q, D)}{\sum_{\tau \in T} \text{TagScore}_M(\tau, q, D)} \tag{10}$$

$$\text{TagScore}_M(t, q, D) = \sum_{d \in D} \text{DocSim}_M(q, d, D) \cdot isTag(t, d) \tag{11}$$

$\text{TagScore}_M(t, q, D)$ calculates the tag score determining how relevant the tag t is to document query q. This score can be any real non-negative number. $\text{DocSim}_M(q, d, D)$ measures the similarity between two documents, i.e. q and d, given a document corpus D and returns a similarity measure ranging between [0,1]. isTag(t, d) is a binary function that returns 1 if $t \in$ d.tags and 0 otherwise. We propose two approaches to compute the document similarity: *Term Frequency-Inverse Document Frequency (TF-IDF)* based ($\text{DocSim}_{\text{TF-IDF}}(q, d, D)$) and *Topic Modeling (TM)* based ($\text{DocSim}_{TM}(q, d, D)$). These two approaches are described in the next subsections.

5.1 TF-IDF-based document similarity

The TF-IDF-based document similarity scoring function, $\text{DocSim}_{\text{TF-IDF}}(q, d, D)$, relies on the TF-IDF principle discussed in Sect. 2.2. The function aims to quantify the content similarity based on term overlap between two documents. To compute the IDF part of the equation, all the documents in D are first indexed. Hence the training phase (preprocess) involves indexing all the documents. The similarity between the query q and a source document d is then computed using $\text{DocSim}_{\text{TF-IDF}}(q, d, D)$ as defined in Eq. 6.

5.2 TM-based document similarity

The TM-based document similarity, $\text{DocSim}_{TM}(q, d, D)$, utilizes topic distributions of the documents computed by the LDA algorithm described in Sect. 2.3. The algorithm further extracts the document semantics using its topic distribution. With this knowledge in mind, one can measure the semantic similarity between two documents by quantifying the similarity between their topic distributions. Indeed, our proposed TM-based algorithm transforms the topic distribution of a document into a numerical vector, wherein cosine similarity is used to compute the topic similarity between two documents using Eq. 9.

[13] http://snowball.tartarus.org/algorithms/english/stemmer.html.

6 Evaluation and results

We evaluate our methods using the tag prediction protocol. We artificially create a test query document by removing the tags from an annotated document. The task is to predict the removed tags. There are two reasons behind the choosing of this evaluation scheme:

1. The evaluation can be done fully automatically. Since our datasets are large, manual evaluation (i.e. having human identify whether a recommended tag is relevant or not) would be infeasible.
2. The evaluation can be done against the existing gold standard established (manually tagged) by expert annotators (i.e. data collectors, project principal investigators, etc.) who have good understanding about the data, while manual evaluation by individuals who are not familiar with the data could lead to evaluation biases.

We evaluate our TF-IDF- and TM-based algorithms against the baseline KEA document annotation algorithm with controlled vocabulary. In our setting, the tag library is used as the vocabulary by the KEA algorithm. The document-wise tenfold cross-validation is performed, where each dataset is first split into 10 equal subsets, and for each fold $i \in \{1, 2, 3, \ldots, 10\}$ the subset i is used for the testing set, and the other 9 subsets are combined and used as the source (training set). The results of each fold are summed up and the averages are reported.

For the TF-IDF-based algorithm, we use LingPipe[14] to perform the indexing and calculating the TF-IDF-based similarity. For the TM-based algorithm, the training process involves modeling topics from the source using LDA algorithm as discussed in Sect. 2.3. We use the Stanford Topic Modeling Toolbox[15] with the collapsed variational Bayes approximation [2] to learn topics in the source documents. For each document we generate uni-grams, bi-grams, and tri-grams, and combine them to represent the textual content of the document. The algorithm takes two input parameters: the number of topics to be identified and the maximum number of training iterations. After some experiments on varying the two parameters, we fix them at 300 and 1000, respectively. The inference method proposed by Asuncion et al. [2] is used to assign a topic distribution to a given document. The evaluation is done on a Windows 7 PC with Intel Core i7 2600 CPU 3.4 GHz and 16GB of RAM.

6.1 Evaluation metrics

This section presents the evaluation metrics used in our tasks, including precision, recall, F1, Mean Reciprocal Rank (MRR), and Binary Preference (Bpref). These metrics, when used in combination, have shown to be effective for evaluation of recommending systems [10,15,38].

6.1.1 *Precision, recall, and F1*

Precision, recall, and F1 (F-measure) are well-known evaluation metrics in information retrieval literature [16]. For each document query in the test set, we use the original set of tags as the ground truth T_g. Assume that the set of recommended tags are T_r, so that the correctly recommended tags are $T_\mathrm{g} \bigcap T_\mathrm{r}$. Precision, recall, and F1 measures are defined as follows:

$$\text{precision} = \frac{|T_\mathrm{g} \bigcap T_\mathrm{r}|}{|T_\mathrm{r}|}, \quad \text{recall} = \frac{|T_\mathrm{g} \bigcap T_\mathrm{r}|}{|T_\mathrm{g}|},$$

$$F1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

In our experiments, the number of recommended tags ranges from 1 to 30. It is wise to note that better tag recommendation systems tend to rank correct tags higher than the incorrect ones. However, the precision, recall, and F1 measures do not take ranking into account. To evaluate the performance of the ranked results, we employ the following evaluation metrics.

6.1.2 *Mean reciprocal rank*

Mean reciprocal rank (MRR) measure takes ordering into account [34]. It measures how well the first correctly recommended tag is ranked. The reciprocal rank of a query is the multiplicative inverse of the rank of the first correctly recommended tag. The mean reciprocal rank is the average of the reciprocal ranks of the results of the query set Q. Formally, given a testing set Q, let rank_q be the rank of the first corrected answer of the query $q \in Q$, then MRR of the query set Q is defined as follows:

$$\text{MRR} = \frac{1}{|Q|} \sum_{q \in Q} \frac{1}{\text{rank}_q}$$

If the set of recommended tags does not contain a correct tag at all, $\frac{1}{\text{rank}_q}$ is defined to be 0.

6.1.3 *Binary preference*

Binary preference (Bpref) measure considers the order of each correctly recommended tag [5]. Let S be the set of recommended tags by the system, R be the set of correct tags (Note that it is not necessary that $R \subseteq S$), $r \in R$ be a correct recommendation, and $i \in S - R$ be an incorrect recommendation. Bpref is defined as follows:

[14] http://alias-i.com/lingpipe/.

[15] http://nlp.stanford.edu/software/tmt/tmt-0.4/.

$$\text{Bpref} = \frac{1}{|R|}\sum_{r\in R} 1 - \frac{|i \text{ ranked higher than } r|}{|S|}$$

Bpref can be thought of as the inverse of the fraction of irrelevant tags that are recommended before relevant ones. Bpref and mean average precision (MAP) are similar when used with complete judgments. However, Bpref normally gives a better evaluation when used in a system with incomplete recommendations.

6.2 Results

Figures 2, 3, and 4 plot the precision@K, recall@K, F1@K, respectively, evaluated at the top K tags recommended by the proposed TF-IDF- and TM-based algorithms against the baseline KEA algorithm on each dataset. Figure 5 summarizes the precision vs. recall on each dataset.

According to the results, our proposed algorithms outperform the baseline KEA algorithm on the DAAC and KNB datasets (TM-based approach outperforms at every K and TF-IDF-based approach outperforms at larger K). This is because the tags used to annotate DAAC and KNB documents are drawn from the libraries of topics. Hence, there is a high chance that a tag is reused for multiple times, resulting in high tag utilization. Since our algorithms give higher weight to tags that have been used frequently, datasets with high tag utilization (such as DAAC and KNB) tend to benefit from our algorithms.

However, our proposed algorithms perform worse than the baseline on the DRYAD dataset. This is because tags in each DRYAD document are manually made up at the curation process. Manually making up tags for each document results in a large size of tag library where each tag is used only a few times, leading to the low tag utilization. Datasets with low tag utilization would not benefit from our proposed algorithms since the probability distribution given to the tags tends to be uniform and not very discriminative.

All the algorithms perform poorly on the TreeBASE dataset. This is because TreeBASE documents are very sparse (some do not even have textual content) and have very few tags. From the dataset statistics, each document on the TreeBASE dataset has only 11 words and only 0.7 tags on average. Such sparse texts lead to weak relationship when finding textually similar documents in the TF-IDF-based approach, and the poor quality of the topic model used by the TM-based

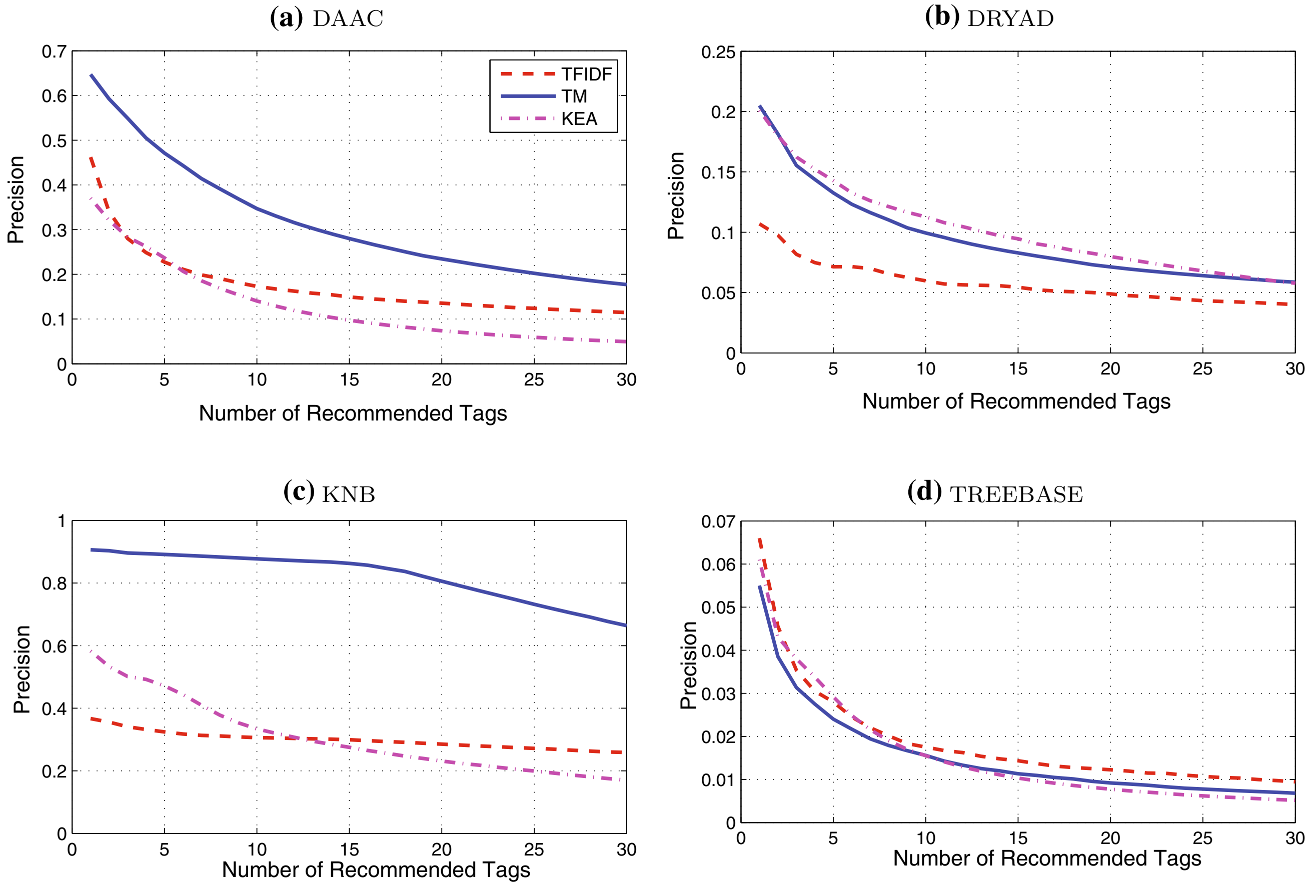


Fig. 2 Precision of the TF-IDF, TM, KEA (*baseline*) algorithms on the four datasets

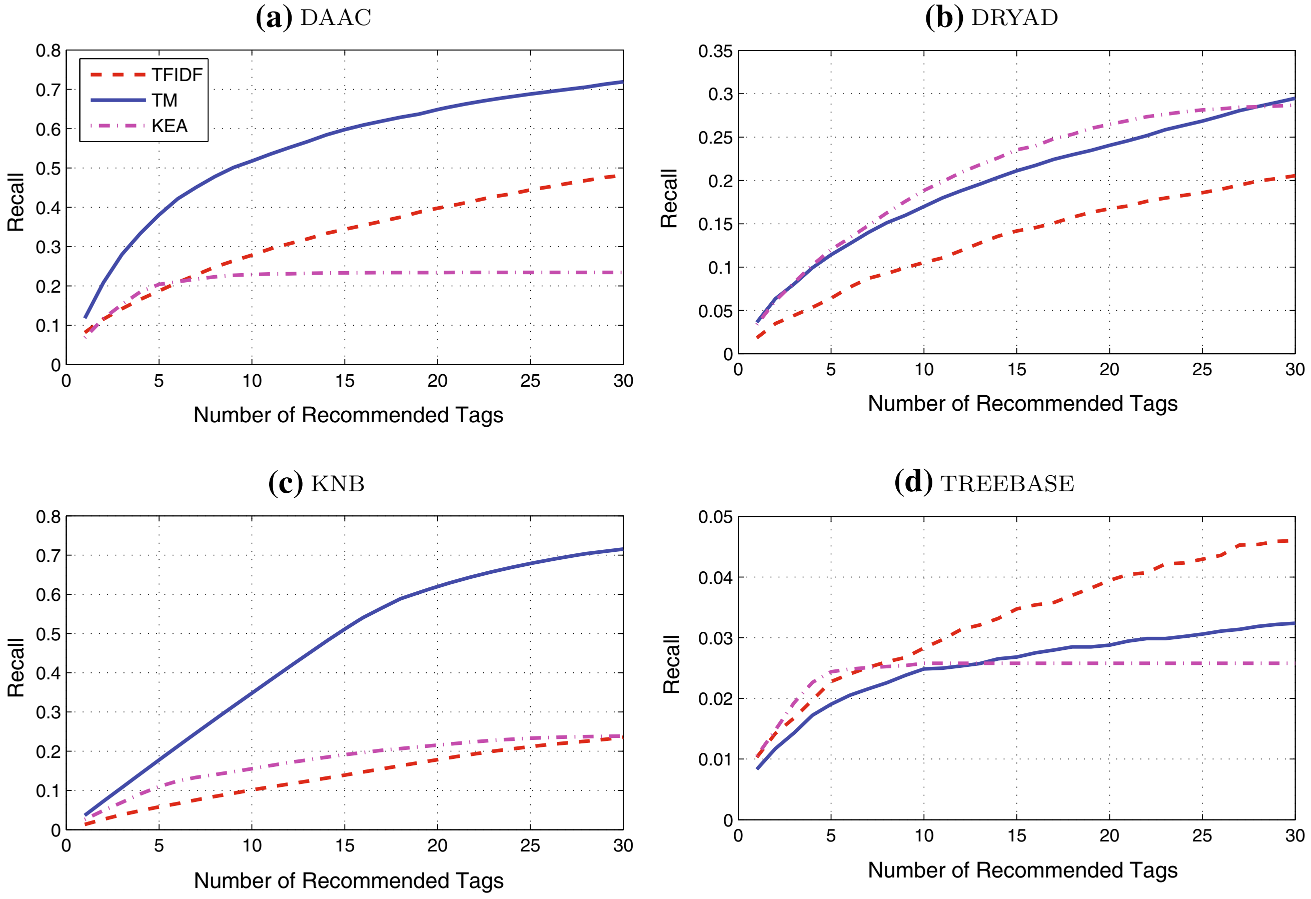


Fig. 3 Recall of the TF-IDF, TM, KEA (*baseline*) algorithms on the four datasets

approach. The small number of tags per document makes it even harder to predict the right tags.

Table 2 lists the MRR, BPref, average learning time (in seconds) per fold, and average testing time (in seconds) per fold of the proposed TF-IDF- and TM-based algorithms against the baseline KEA algorithm on each dataset. MRR quantifies how the first correct recommendation is ranked. In terms of MRR, our TM-based algorithm performs the best on the DAAC and KNB datasets, TF-IDF-based algorithm performs the best in the TreeBASE dataset, and the KEA algorithm performs the best on the DRYAD dataset. The TM-based algorithm achieves notable MRR scores of 0.75 and 0.92 on the DAAC and KNB datasets, respectively, and outperforming the baseline by 47.70 and 33.22 %, respectively.

Bpref measures the ranking of all the correctly recommended keywords. In terms of Bpref, our TM algorithm performs the best on the DAAC, DRYAD, and KNB datasets with the Bpref scores of 0.90, 0.49, and 0.91 respectively. The TF-IDF-based algorithm performs the best on the TreeBASE dataset. Similar to the MRR results, notable BPref scores are achieved by the TM-based algorithm on the DAAC and KNB datasets, outperforming the baseline by 285.32 and 274.33 % respectively.

Table 3 shows sample recommended tags by our proposed TF-IDF/TM based algorithms and the baseline KEA algorithm to the DAAC metadata record titled "*ISLSCP II IGBP DISCOVER AND SIB LAND COVER, 1992-1993*"[16], against the 15 actual ground-truth tags associated with the record. Our TM-based algorithm performs well on this particular example by capturing all the actual tags within the top 15 recommended tags.

7 Discussion

This section provides additional discussions about the proposed algorithms.

7.1 TM- vs. TF-IDF-based approaches

According to the results, our TM-based approach performs better than the TF-IDF-based approach on DAAC, DRYAD, and KNB datasets, in terms of precision, recall, and F1 measure, while the TF-IDF-based approach performs better on the TreeBASE dataset. Since the only difference between

[16] http://daac.ornl.gov/cgi-bin/dsviewer.pl?ds_id=930.

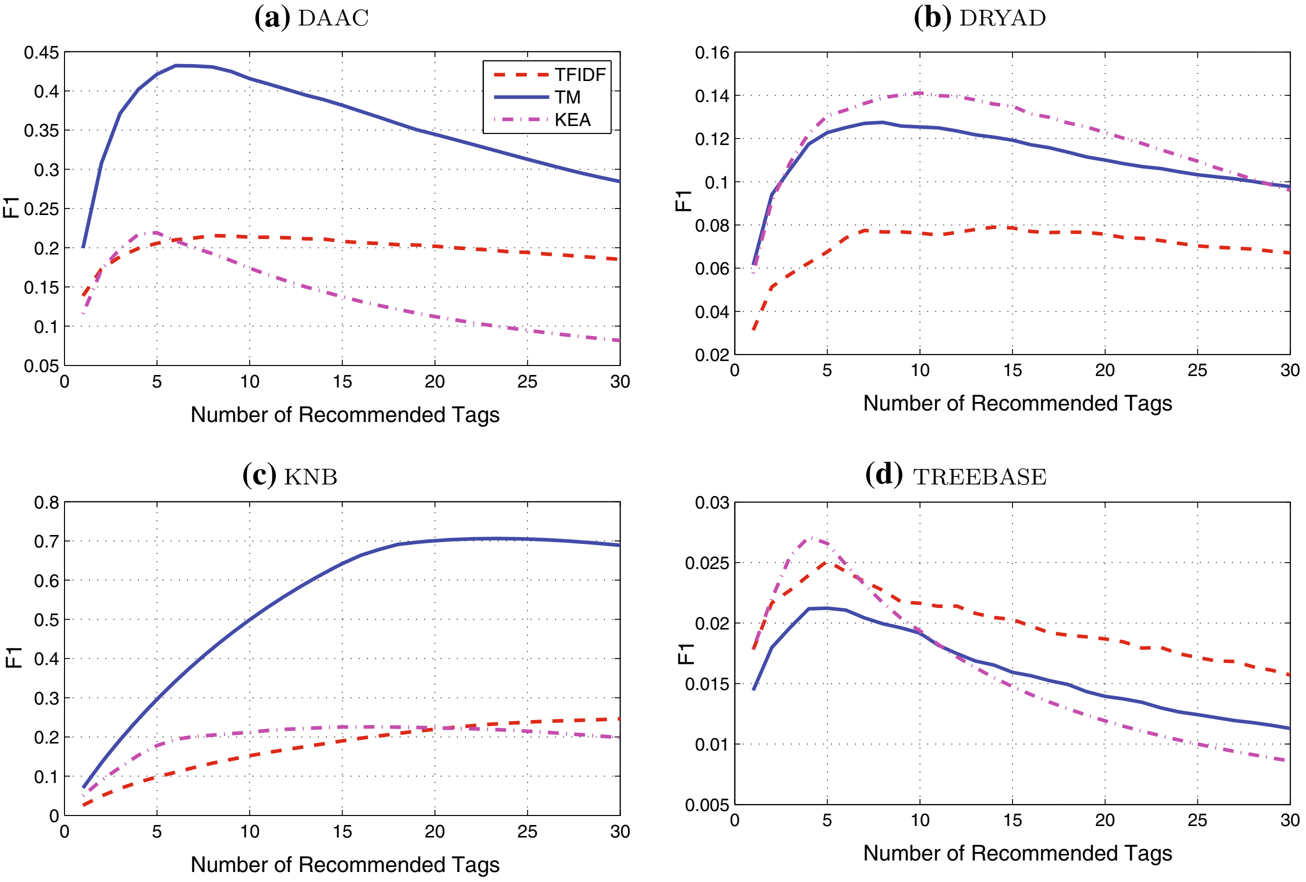


Fig. 4 F1 of the TF-IDF, TM, KEA (*baseline*) algorithms on the four datasets

the two proposed methods is the document similarity function DocSim(q, d, D), which computes the similarity score between the query document q and a source document $d \in D$, the analysis on the differences between the two document similarity measures could provide explanation about the performance difference.

The TF-IDF document similarity quantifies the cosine similarity between two TF-IDF vectors representing the two documents. Loosely speaking, the TF-IDF document similarity measures the quantity of term overlap, where each term has a different weight, in the two documents.

The TM-based approach first derives a set of topics from the document source, each of which is represented by a distribution of terms. The ranked terms in each topic bare coherent semantic meanings. Table 4 provides an example of the top 10 terms of each of the sample 9 topics derived from the DAAC dataset using the LDA algorithm with 300 topics and 1000 iterations. Once the set of topics has been determined, a document is assigned a distribution of topics using the inference algorithm mentioned in Sect. 2.3. The TM document similarity then measures the cosine similarity between the topic distribution vectors representing the two documents. Loosely speaking, the TM document similarity quantifies the topic similarity between the two documents.

The performance difference of both the proposed methods could be impacted by the semantic representation of each document. It is evident from the experimental results on the DAAC, DRYAD, and KNB datasets that representing a document with a mixture of topics leads to more accurate semantic similarity interpretation, resulting in better recommendation. However, the reason why the TM-based approach performs worse than the TF-IDF-based approach on the TreeBASE dataset could be that the documents in such a dataset are very sparse (each TreeBASE document has only 11 words on average). Such sparsity could lead to a poor set of topics, consisting of idiosyncratic word combinations.

Hence we recommend the TM-based algorithm for datasets whose documents are rich in textual content, and the TF-IDF-based algorithm approach for those with textually sparse documents.

7.2 Limitations

Regardless of the promising performance, our proposed document annotation algorithms may face the following limitations:

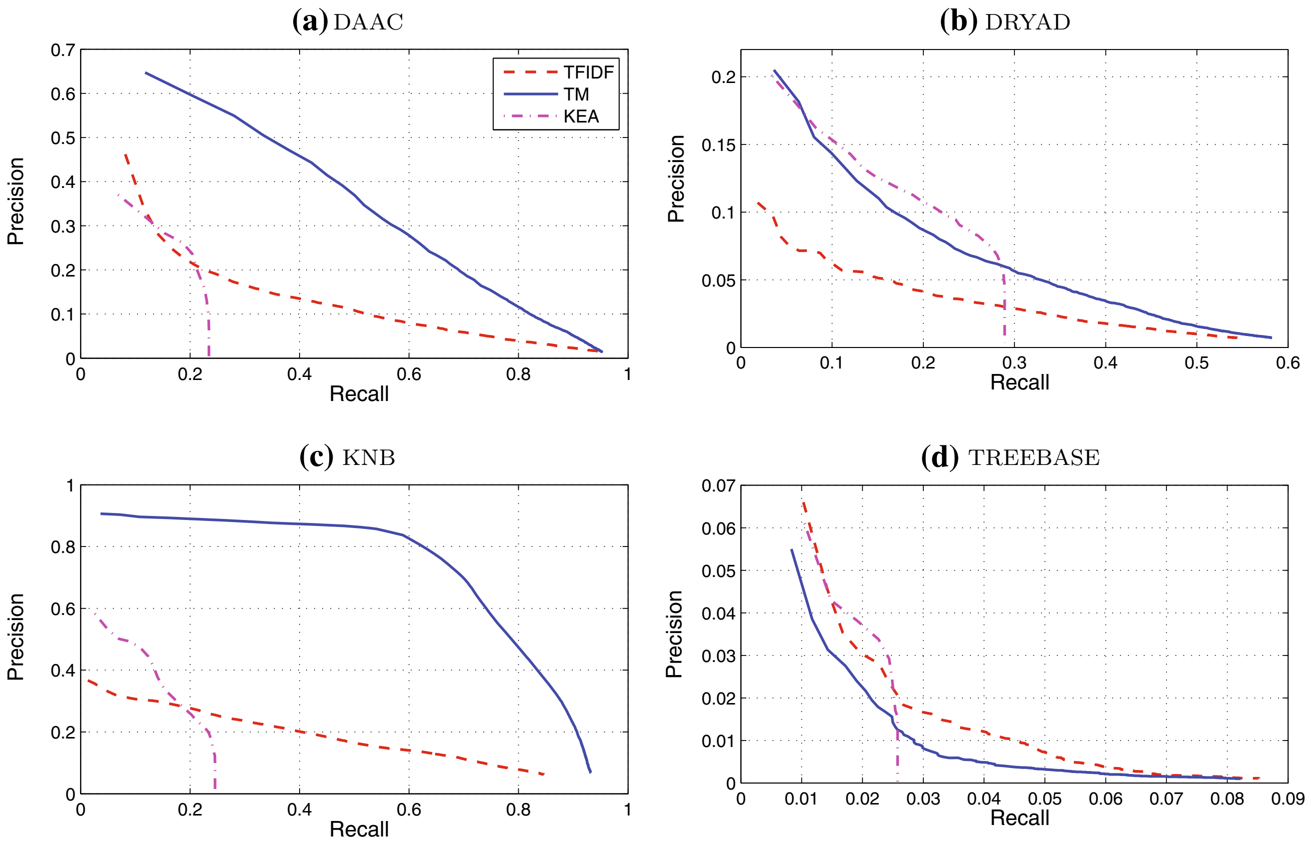


Fig. 5 Precision vs. recall of the TF-IDF, TM, KEA (*baseline*) algorithms on the four datasets

Table 2 MRR, Bpref, average learning time (ALT) and average test time (ATT) of TF-IDF, TM, KEA (*baseline*) algorithms on the four datasets

Dataset	Method	MRR	Bpref	ALT (s)	ATT (s)
DAAC	TFIDF	0.5649	0.8183	5.29	0.99
	TM	**0.7546**	**0.9005**	2430.82	49.63
	KEA	0.5109	0.2337	75.45	6.50
DRYAD	TFIDF	0.2022	0.4404	6.32	1.19
	TM	0.3264	**0.4934**	4486.09	83.76
	KEA	**0.3423**	0.2851	102.59	8.85
KNB	TFIDF	0.4944	0.6659	6.06	1.23
	TM	**0.9226**	**0.9100**	1159.81	49.60
	KEA	0.6823	0.2431	64.11	5.61
TREEBASE	TFIDF	**0.0893**	**0.0695**	6.15	1.08
	TM	0.0750	0.0636	401.50	16.99
	KEA	0.0745	0.0257	6.26	1.04

Bold values represent the best performance achieved by the corresponding methods in each dataset and evaluation metric

1. The proposed algorithms rely on the existence of a good document source (training set). The quality of the resulting annotation directly reflects the quality of the annotation of each document in the training data. Fortunately, the current ONEMercury system only retrieves the metadata from the archives wherein each metadata record is manually and carefully annotated by principal investigators and data managers. In the future, however, the system may expand to collect metadata from sources in which the metadata records may have poor or no annotation. Such problems urge the need for a method that allows the automatic annotator trained with a high-quality training dataset to annotate the documents in different datasets. Indeed, we briefly discuss the possibility of

Table 3 Comparison of the recommended keywords by the TF-IDF, TM, and KEA (baseline) algorithms on a sample document "*ISLSCP II IGBP DISCOVER AND SIB LAND COVER, 1992–1993*"

Actual Tags	TFIDF	TM	KEA(Baseline)
1. albedo	1. **field investig**	1. **land cover**	1. model
2. land cover	2. analysi	2. **modi moder resolut imag spectroradiomet**	2. geograph distribut
3. veget cover	3. **land cover**	3. **terra morn equatori cross time satellit**	3. classif
4. veget index	4. comput model	4. **field investig**	4. lba
5. leaf area meter	5. **reflect**	5. **veget cover**	5. amazonia
6. terra morn equatori cross time satellit	6. **veget cover**	6. **reflect**	6. area
7. noaa nation ocean amp amp atmospher administr	7. biomass	7. **veget index**	7. south america
8. plant characterist	8. primari product	8. **leaf characterist**	8. ecolog
9. steel measur tape	9. **steel measur tape**	9. **canopi characterist**	9. **reflect**
10. canopi characterist	10. weigh balanc	10. **plant characterist**	10. calibr
11. modi moder resolut imag spectroradiomet	11. precipit amount	11. **albedo**	11. field investig
12. leaf characterist	12. **canopi characterist**	12. **steel measur tape**	12. speci
13. avhrr advanc high resolut radiomet	13. **leaf characterist**	13. **avhrr advanc high resolut radiomet**	13. factor
14. field investig	14. water vapor	14. **noaa nation ocean amp amp atmospher administr**	14. sequenc
15. reflect	15. quadrat sampl frame	15. **leaf area meter**	15. hawaiian island
	16. rain gaug	16. analysi	16. genera
	17. surfac air temperatur	17. comput model	17. fern
	18. air temperatur	18. noaa	18. systemat
	19. meteorolog station	19. avhrr	19. **steel measur tape**
	20. human observ	20. popul distribut	20. correl

The first column lists the actual tags. The bold, underlined terms are correctly recommended items

Table 4 Top 10 terms in sample 9 topics derived using LDA algorithm from the DAAC dataset

Topic1	Topic2	Topic3	Topic4	Topic5	Topic6	Topic7	Topic8	Topic 9
amazon	pine	aboveground	park	carbon dioxide	soil moisture	plant	bigfoot	environment
river	team	year	mongu	ssa	site average	water potential	modi product	radar
basin	black spruce	mortal	safari	energy	moisture bett	determin	evergreen	mosaic
ecology	chamber	woody biomass	zambia	exchange	neutron	leaf optic	land cover	geotiff format
floodplain	methane	hardwood	wet season	soil temperature	airborn	radiometer	bigfoot project	band
band	team collect	brown	photosynthetic	team	fife experiment	plant water	nasa	backscatter image
inundate	chamber flux	eastern	kalahari	tower flux	gamma	grass	flux tower	synthesize
meter	trace gas	softwood	botswana	vapor	gravimeter	leaf water	grassland	topograph
scale	plant	schroeder	store	water vapor	water content	leaf tissue	evergreen needleleaf	srtm
mosaic	spruce	commercial	activity	flux measure	gas flux	summer	ecology	develop

applying the proposed method for cross-archive annotation in Sect. 7.5.

2. Our TM-based algorithm needs to model topics from scratch every time a significant amount of new documents are added to the training corpus, so that the modeled topics can reflect the new documents added. Since our TM-based algorithm utilizes the traditional LDA algorithm to model topics, wherein incremental training is not a feature, we plan to explore methods such as [1] and [11] which may enable our algorithm to adaptively model the topics from a dynamic corpus.
3. Regardless of the promising performance of our proposed TM-based algorithm, the scalability can be an issue when it comes to mining topics from a larger corpus of documents. The scalability issues of our TM-based algorithm is discussed in detail in the next subsection.

7.3 Scalability of the TM approach

Scalability issues should be taken into account since the algorithms will eventually be incorporated as part of the ONEMercury system, which currently hosts much larger datasets than the ones we use in the experiments. Since theoretical time and space complexities of the underlying LDA algorithm have been extensively investigated (see [22]), we instead focus on the scalability issues from the practical point of view. This section discusses two scalability issues presented in the TM-based algorithm: the increase in number of topics and the increase in size of the corpus.

We examine the scalability issues of the proposed TM-based algorithm on the KNB dataset, using the Stanford Topic Modeling Toolbox with collapsed variational Bayes approximation and fixed 1000 iterations, on the same machine we use for earlier experiments.

As the data grow larger, new topics emerge, urging the need for a new model that captures such increasing variety of topics. Figure 6 plots the training time (in seconds) as a function of number of topics. The training time grows approximately linearly with the number of topics up to 400 topics. The program runs out of physical memory, however, at 500 topics, leading to a dramatic increase in the training time. Hence, this study points out that a more memory-efficient topic model algorithm should be explored.

Another scalability concern lies with the projected increase in the number of training documents. Figure 7 shows the training time of the TM-based algorithm as the number of documents increases. The results also show a linear scale with the number of training documents. Note that the experiment is only done with up to 1000 documents, while there are roughly 47 thousands, and definitely increasing in the future, metadata records in the current system. Even with the current size of the ONEMercury repository, the algorithm would take approximately 5.3 h to model topics, which is not feasible in practice. Hence a large-scale parallel algorithm such as MapReduce [6] should be investigated.

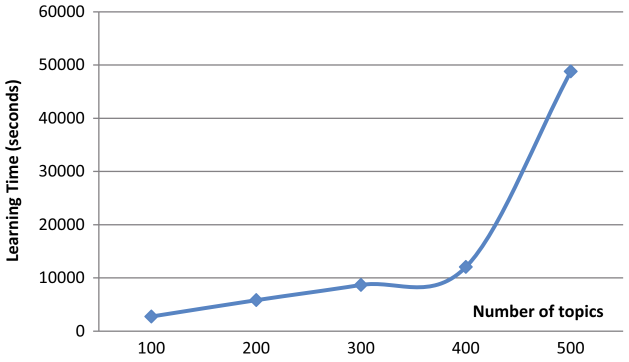


Fig. 6 Learning time in seconds of the TM-based algorithm as a function of number of topics

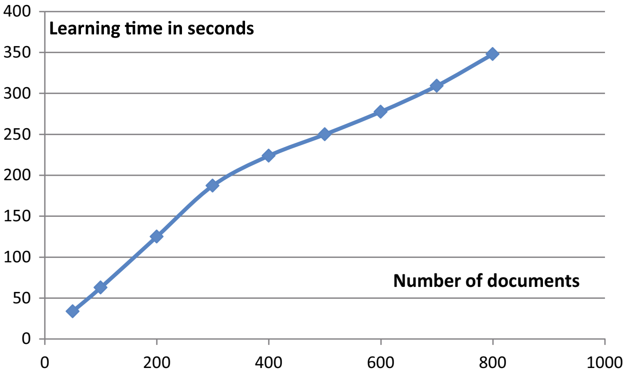


Fig. 7 Learning time in seconds of the TM-based algorithm as a function of numbers of training documents

7.4 Employing topic coherence to find optimum numbers of topics

Multiple studies on topic modeling have shown that the coherence of the term distribution in each topic has a direct impact on the effectiveness of the learned topics in various applications [3,7,9,35]. Newman et al. [21] defined the coherence of a topic as the ability to be interpreted by human as a semantically meaningful topic. Since our TM-based method utilizes LDA to learn topical knowledge from the source documents to compute topical similarity between documents, the coherence of the learned topics could have an impact on the relevance of the recommended tags.

In this section, the coherence of topics learned from each dataset is investigated. The results do not only shed light on the quality of the learned topics, but also help determine the optimal number of topics to be learned from each archive. Too few topics would typically result in very broad topics; while too many topics will result in random, meaningless topics that pick out idiosyncratic word combinations [24].

Newman et al. [21] proposed a set of schemes for automatic evaluation of topic coherence, divided into three groups: WordNet, Wikipedia, and Google search engine-based methods. For our topic coherence analysis, we adopt the similar evaluation scheme as their Wikipedia-based method using pair-wise mutual information (PMI) as the word-pair scoring function, since this scheme was reported the most accurate in the authors' work.

Since we aim to find the optimal number of topics to learn from each archive, the aggregate topic coherence score is calculated for each topic set. In particular, let $Z_T = \{z_1, z_2, \ldots, z_T\}$ be the set of T learned topics. We aim to calculate the aggregate topic coherence (ATC) score for the topic set Z_T by taking the arithmetic *mean* and *median* of the coherence scores of all the topics in Z_T, as follows:

$$\mathrm{ATC}_{\mathrm{mean}}(Z_T) = \mathrm{mean}\{C(z_i); i \in 1 \ldots T\} \tag{12}$$

$$\mathrm{ATC}_{\mathrm{median}}(Z_T) = \mathrm{median}\{C(z_i); i \in 1 \ldots T\} \tag{13}$$

$C(z)$ is a coherence score of the topic z and is calculated as follows:

Let z be a topic, and $W_{10} = \{w_1, \ldots, w_{10}\}$ be the top 10 words in z. Then, the coherence score of a topic z is the average of pair-wise mutual information (PMI) scores of all possible unique pairs of the words in W_{10}.

$$C(z) = \mathrm{mean}\{\mathrm{PMI}(w_i, w_j); ij \in 1 \ldots 10; i < j\} \tag{14}$$

$$\mathrm{PMI}(w_i, w_j) = \log\left(\frac{p(w_i, w_j)}{p(w_i) \cdot p(w_j)}\right) \tag{15}$$

$p(w_i)$ and $p(w_j)$ are calculated using the portions of documents that contain at least one occurrence of w_i and w_j, respectively. $p(w_i, w_j)$ is the portion of the documents that contain both w_i and w_j. Mathematically, let D be the document collection, $D(w) \in D$ be the set of documents containing at least one occurrence of w.

$$p(w_i) = \frac{|D(w_i)|}{|D|}, \quad p(w_j) = \frac{|D(w_j)|}{|D|} \tag{16}$$

$$p(w_i, w_j) = \frac{|D(w_i) \cap D(w_j)|}{|D|} \tag{17}$$

Instead of using Wikipedia articles as the external knowledge source as in [21], we use the documents in the datasets as the external knowledge. This is because, most metadata records in our datasets are from very specific subfields of ecological and environmental sciences, which Wikipedia articles do not well cover. Plus, these metadata records contain many technical and scientific keywords which are not normally used in general encyclopedias. For each dataset, a set of randomly chosen 1000 documents is used to model the topics. The numbers of topics, T, are varied by 50 during 0–600 topics, and by 100 during 600–2000 topics. At each T, the LDA algorithm is run with 1000 iterations to learn a set of T topics, Z_T. Then, the *mean* and *median* aggregate topic coherence scores are calculated for each Z_T.

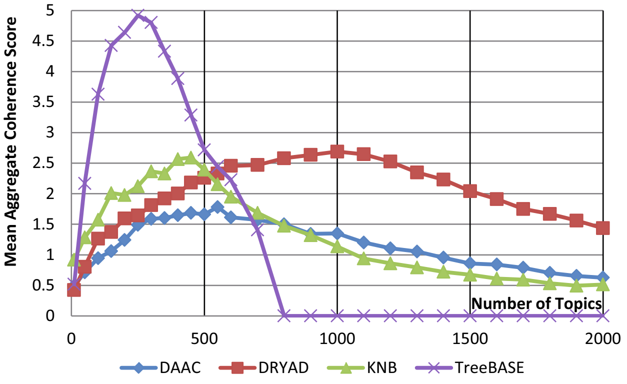


Fig. 8 The *mean* aggregate topic coherence scores of the four datasets as a function of number of topics

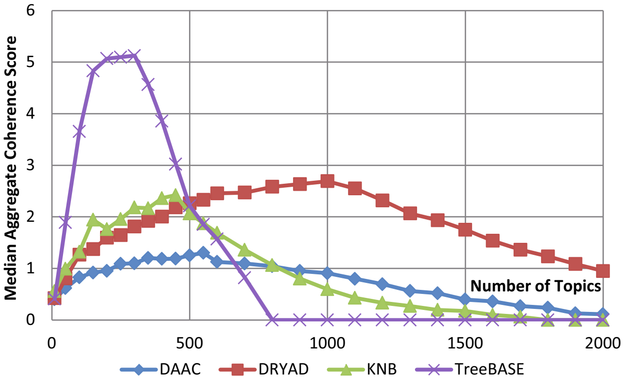


Fig. 9 The *median* aggregate topic coherence scores of the four datasets as a function of number of topics

Figures 8 and 9 plot the *mean* and *median* of the aggregate topic coherence scores, respectively, of each dataset as a function of number of topics. According to Fig. 8, the optimal numbers of topics to be learned from datasets DAAC, DRYAD, KNB, and TreeBASE are 550, 1000, 450, and 250, respectively. Note that, according to Table 1, the approximated data sizes for DAAC, DRYAD, KNB, and TreeBASE used in this analysis are 104,261, 129,926, 63,324, and 11,405 words, respectively. Surprisingly, there is a correlation between the optimal numbers of topics and the sizes of the document collections. An explanation of this phenomenon could be that richer archives tend to have more content, and hence are composed by more topical subjects. It is also interesting to note that the effect of *too many topics* is apparent in the TreeBASE dataset, where the mean aggregate topic coherence scores significantly drop closed to zero after 700 topics. This is because the size of the TreeBASE document collection used in this analysis is so small that additional topics beyond 700 topics become random and spurious, hence impeding overall quality of the learned topics.

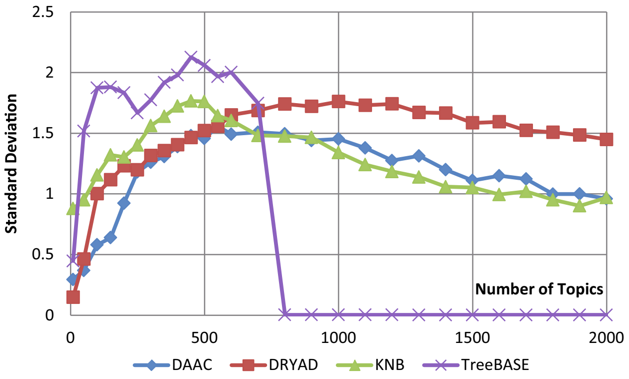
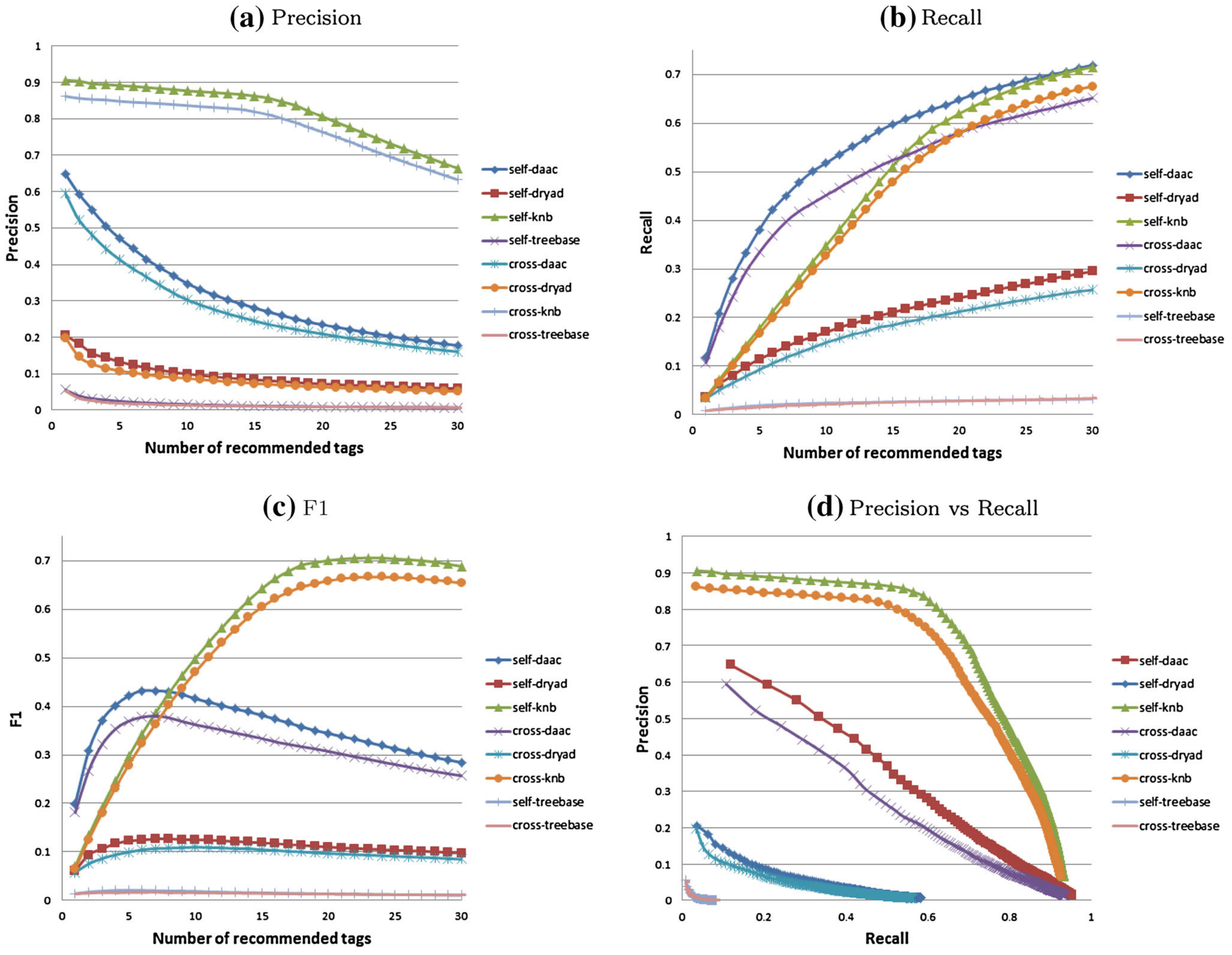


Fig. 10 The *standard deviation* of the aggregate topic coherence scores of the four datasets as a function of number of topics

Figure 10 plots the *standard deviation* of the topic coherence scores of the four datasets at different numbers of topics. Interestingly, the standard deviation directly correlates with the aggregate topic coherence scores in Figs. 8 and 9. This is because, with too small numbers of topics (less than the optimum), the learned topics tend to be general resulting in similar semantics across all the topics, leading to lower standard deviation. At the other extreme, larger numbers of topics (beyond the optimum point) may cause all the learned topics to be equally random, hence lower standard deviation.

7.5 Experiments on cross-archive annotation

In most cases, the documents used to model the annotator are selected from the same archive as the target document (self-archive annotation), with the intuition that documents in the same archive tend to have similar topical composition. However, the annotator modeled from multiple archives may also be useful. This method is call cross-archive annotation and can provide the following benefits:


(a) Precision

(b) Recall

(c) F1

(d) Precision vs Recall

Fig. 11 Precision, recall, F1, and precision-vs.-recall of the TM-based method performed on different data sets and source selection modes. **a** Precision, **b** recall, **c** F1, **d** precision vs. recall

1. **Mitigating the cold start problem.** The cross-archive annotation can solve the cold-start problem where the documents needed to be annotated do not have an associated richly annotated source archive to train the annotator.
2. **Introducing new but relevant topical knowledge.** Different archives bare a wide variety of topical knowledge and annotation. Modeling an annotator from multiple sources hence would introduce new concepts and tags to the annotator.

To investigate the possibility of applying the proposed methodology on cross-archive annotation, an experiment is conducted using the TM-based method to compare the performance between self-archive and cross-archive annotation. For the self-archive annotation evaluation, the documents in the training set are selected from the same dataset as the target document; while the cross-archive annotation evaluation combines the training documents from all the four datasets together. We evaluate the proposed TM-based algorithm with different source modes using document-wise tenfold cross-validation, where each data set is split into ten equal subsets, and for each fold $i \in \{1, 2, 3, \ldots, 10\}$ the subset i is used for the testing set, and the other nine subsets are combined and used as the source (training set).

Figure 11 shows the comparison of precision, recall, F1, and precision vs. recall of the self- and cross-archive evaluations on the test documents from the four datasets. Interestingly, the cross-archive performance is worse than the self-archive evaluation for all the four test datasets. This is because the prediction protocol is used as the evaluation criteria, where we try to predict the pre-existing tags of the test documents. As a result, the tags from different tag vocabularies may be unknown to the target documents. Hence, even though cross-archive annotation has the potential to bring a new variety of relevant annotations to the target documents, the evaluation criteria used here are too strict (due to being automatic), and hence an expert evaluation where professionals manually review the results of the annotation would be needed to enhance the evaluation of the cross-archive annotation methodology.

8 Conclusion and future work

This paper presents a set of algorithms for automatic annotation of metadata. We are motivated by the real-world problems faced by ONEMecury, a search system for environmental science metadata harvested from multiple data archives. One of the important problems includes the different levels of curation of metadata from different archives, which means that the system must automatically annotate metadata records which are poorly annotated. We treat each metadata record as a tagged document, and then transform the problem into the tag recommendation problem with a controlled tag library.

We propose two algorithms for tag recommendation, one based on term frequency-inverse document frequency (TF-IDF) and the other based on topic modeling (TM) using the Latent Dirichlet Allocation. The evaluation is done on four different datasets of environmental science metadata using the tag prediction evaluation protocol, against the well-known KEA document annotation algorithm. The results show that our TM-based approach yields better results on datasets characterized by high tag utilization and rich in textual content such as DAAC and KNB than those which do not (i.e. DRYAD and TreeBASE), though with the cost of longer learning times. The scalability issues of the TM-based algorithm necessitate investigation into more memory-efficient and scalable approaches. Finally, future steps could be implementing an automatic metadata annotation algorithm on the ONEMercury search service or exploring online tagging [23].

Acknowledgments We gratefully acknowledge useful comments from Natasha Noy, Jeffery S. Horsburgh, and Giri Palanisamy. This work has been supported in part by the National Science Foundation (DataONE: Grant #OCI-0830944) and Oak Ridge National Laboratory Contract No. De-AC05-00OR22725 of the US Department of Energy.

References

1. AlSumait, L., Barbar, D., Domeniconi, C.: On-line lda: adaptive topic models for mining text streams with applications to topic detection and tracking. In: IEEE Computer Society ICDM, pp. 3–12 (2008)
2. Asuncion, A., Welling, M., Smyth, P., Teh, Y.W.: On smoothing and inference for topic models. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI '09, pp. 27–34. AUAI Press, Arlington (2009)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
4. Bron, M., Huurnink, B., de Rijke, M.: Linking archives using document enrichment and term selection. In: Proceedings of the 15th International Conference on Theory and Practice of Digital Libraries: Research and Advanced Technology for Digital Libraries, TPDL'11, pp. 360–371 (2011)
5. Buckley, C., Voorhees, E.M.: Retrieval evaluation with incomplete information. In: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '04, pp. 25–32. ACM, New York (2004)
6. Dean, J., Ghemawat, S.: Mapreduce: simplified data processing on large clusters. Commun. ACM **51**(1), 107–113 (2008)
7. Deerwester, S.C., Dumais, S.T., Landauer, T.K., Furnas, G.W., Harshman, R.A.: Indexing by latent semantic analysis. JASIS **41**(6), 391–407 (1990)
8. Heymann, P., Ramage, D., Garcia-Molina, H.: Social tag prediction. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '08, pp. 531–538. ACM, New York (2008)
9. Hofmann, T.: Unsupervised learning by probabilistic latent semantic analysis. Mach. Learn. **42**(1–2), 177–196 (2001)

10. Huang, W., Kataria, S., Caragea, C., Mitra, P., Giles, C.L., Rokach, L.: Recommending citations: translating papers into references. CIKM '12, pp. 1910–1914. ACM, New York (2012)
11. Iwata, T., Yamada, T., Sakurai, Y., Ueda, N.: Online multiscale dynamic topic models. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '10, pp. 663–672. ACM, New York (2010)
12. Kataria, S., Mitra, P., Bhatia, S.: Utilizing context in generative bayesian models for linked corpus. In: AAAI'10, p. 1 (2010)
13. Krestel, R., Fankhauser, P., Nejdl, W.: Latent dirichlet allocation for tag recommendation. In: Proceedings of the Third ACM Conference on Recommender Systems, RecSys '09, pp. 61–68. ACM, New York (2009)
14. Liu, Z., Chen, X., Sun, M.: A simple word trigger method for social tag suggestion. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11, pp. 1577–1588. Association for Computational Linguistics, Stroudsburg (2011)
15. Liu, Z., Huang, W., Zheng, Y., Sun, M.: Automatic keyphrase extraction via topic decomposition. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP '10, pp. 366–376. Association for Computational Linguistics, Stroudsburg (2010)
16. Manning, C.D., Raghavan, P., Schtze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
17. Medelyan, O., Witten, I.H.: Thesaurus based automatic keyphrase indexing. In: Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '06, pp. 296–297. ACM, New York (2006)
18. Michener, W., Vieglais, D., Vision, T., Kunze, J., Cruse, P., Janée, G.: DataONE: data observation network for earth—preserving data and enabling innovation in the biological and environmental sciences. DLib Mag. **17**(1/2), 1–12 (2011)
19. Mishne, G.: Autotag: a collaborative approach to automated tag assignment for weblog posts. In: Proceedings of the 15th International Conference on World Wide Web, WWW '06, pp. 953–954. ACM, New York (2006)
20. Newman, D., Hagedorn, K., Chemudugunta, C., Smyth, P.: Subject metadata enrichment using statistical topic models. In: Proceedings of the 7th ACM/IEEE-CS Joint Conference on Digital libraries, JCDL '07, pp. 366–375. ACM, New York (2007)
21. Newman, D., Lau, J.H., Grieser, K., Baldwin, T.: Automatic evaluation of topic coherence. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10, pp. 100–108. Association for Computational Linguistics, Stroudsburg (2010)
22. Newman, D., Smyth, P., Welling, M., Asuncion, A.U.: Distributed inference for latent dirichlet allocation. In: Advances in Neural Information Processing Systems, pp. 1081–1088 (2007)
23. Song, Y., Zhuang, Z., Li, H., Zhao, Q., Li, J., Lee, W.-C., Giles, C.L.: Real-time automatic tag recommendation. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '08, pp. 515–522 (2008)
24. Steyvers, M., Griffiths, T.: Probabilistic topic models. Handb. Latent Semant. Anal. **427**(7), 424–440 (2007)
25. Tuarob, S., Pouchard, L.C., Giles, C.L.: Automatic tag recommendation for metadata annotation using probabilistic topic modeling. In: Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL '13, pp. 239–248. ACM, New York (2013)
26. Tuarob, S., Pouchard, L.C., Noy, N., Horsburgh, J.S., Palanisamy, G.: Onemercury: towards automatic annotation of earth science metadata. In: AGU Fall Meeting Abstracts, vol. 1, p. 1482 (2012)
27. Tuarob, S., Pouchard, L.C., Noy, N., Horsburgh, J.S., Palanisamy, G.: Onemercury: towards automatic annotation of environmental science metadata. In: Proceedings of the 2nd International Workshop on Linked Science (2012)
28. Tuarob, S., Tucker, C.S.: Fad or here to stay: predicting product market adoption and longevity using large scale, social media data. In: Proceedings ASME 2013 Internationl Design Engineering Technical Conference Computers and Information in Engineering Conference, IDETC/CIE '13 (2013)
29. Tuarob, S., Tucker, C.S.: Discovering next generation product innovations by identifying lead user preferences expressed through large scale social media data. In: Proceedings ASME 2014 International Design Engineering Technical Conference Computers and Information in Engineering Conference, IDETC/CIE '14 (2014)
30. Tuarob, S., Tucker, C.S.: Automated discovery of lead users and latent product features by mining large scale social media networks. J. Mech. Des. (2015, accepted)
31. Tuarob, S., Tucker, C.S.: Quantifying product favorability and extracting notable product features using large scale social media data. J. Comput. Inf. Sci. Eng. (2015). doi:10.1115/1.4029562
32. Tuarob, S., Tucker, C. S., Salathe, M., Ram, N.: Discovering health-related knowledge in social media using ensembles of heterogeneous features. In: Proceedings of the 22Nd ACM International Conference on Conference on Information and Knowledge Management, CIKM '13, pp. 1685–1690. ACM, New York (2013)
33. Tuarob, S., Tucker, C.S., Salathe, M., Ram, N.: An ensemble heterogeneous classification methodology for discovering health-related knowledge in social media messages. J. Biomed. Inf. **49**, 255–268 (2014)
34. Voorhees, E.M.: The trec-8 question answering track report. In: Proceedings of TREC-8, pp. 77–82 (1999)
35. Widdows, D., Ferraro, K.: Semantic vectors: a scalable open source package and online technology management application. In: LREC (2008)
36. Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C. G.: Kea: practical automatic keyphrase extraction. In: Proceedings of the Fourth ACM Conference on Digital libraries, DL '99, pp. 254–255. ACM, New York (1999)
37. Wu, L., Yang, L., Yu, N., Hua, X.-S.: Learning to tag. In: Proceedings of the 18th International Conference on World Wide Web, WWW '09, pp. 361–370 (2009)
38. Zhou, T., Ma, H., Lyu, M., King, I.: Userrec: A user recommendation framework in social tagging systems. In: Proceedings of AAAI, pp. 1486–1491 (2010)