Hurtado *et al. J Big Data (2016) 3:7*
DOI 10.1186/s40537-016-0039-2

Journal of Big Data

RESEARCH Open Access

Topic discovery and future trend forecasting for texts

Jose L. Hurtado*†, Ankur Agarwal† and Xingquan Zhu†

*Correspondence:
jhurtad2@fau.edu
†The authors Jose L. Hurtado, Ankur Agarwal and Xingquan Zhu are equally contributed.
Department of Computer and Electrical Engineering and Computer Science, Florida Atlantic University, 777 Glades Road, Boca Raton, FL 33431, USA

Abstract

Finding topics from a collection of documents, such as research publications, patents, and technical reports, is helpful for summarizing large scale text collections and the world wide web. It can also help forecast topic trends in the future. This can be beneficial for many applications, such as modeling the evolution of the direction of research and forecasting future trends of the IT industry. In this paper, we propose using association analysis and ensemble forecasting to automatically discover topics from a set of text documents and forecast their evolving trend in a near future. In order to discover meaningful topics, we collect publications from a particular research area, data mining and machine learning, as our data domain. An association analysis process is applied to the collected data to first identify a set of topics, followed by a temporal correlation analysis to help discover correlations between topics, and identify a network of topics and communities. After that, an ensemble forecasting approach is proposed to predict the popularity of research topics in the future. Our experiments and validations on data with 9 years of publication records validate the effectiveness of the proposed design.

Keywords: Machine learning, Text mining, Topic forecast, Topic discovery, Community discovery, Topic identification, Label identification, Cluster labeling, Category labeling, Association rules

Background

Finding topics from a collection of documents can be useful for many real-world applications [2]. For example, by examining recent publications in medical domains [3], we can identify areas which are becoming increasingly important and further predict their trend and popularity in a near future [4]. Similarly, by examining patent applications or online customer review reports, one can find emerging trends of new product designs [5]. By mining financial news and events, it is now possible to predict future trends of the stock market [6]. For all these applications, the underlying technical problem is essentially twofold (1) how to summarize and generate meaningful topics from a set of documents (topic mining or discovery); and (2) how to forecast the trend of topics in the future (topic forecasting).

Finding meaningful topics from a set of documents has been studied in existing research [2, 7, 8]. Existing solutions commonly use text clustering, association rule mining, or latent semantic models for topic discovery. However, for all these methods, each single document is represented as an instance (by using bag-of-word features), without

Springer Open

© 2016 Hurtado et al. This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

considering context information in paragraphs and sentences. In this paper, we propose to use sentence-level association rule mining to discover topics from documents. Our method considers each sentence as a transaction, and keywords within the sentences are regarded as items in the transaction. By exploring keywords, which frequently co-occur, as patterns, our method can maximally preserve context information in the topic mining process. As a result, our topic discovery method is fined-grained to the sentence level context information of the documents for discovering meaningful topics.

Assume a set of topics are discovered from documents, topic forecasting intends to predict the future trend (i.e., popularity) of the topics/patterns, by using temporal correlations of the patterns observed from training data. This problem is essentially a time-series forecasting issue [9], because one can regard the popularity score of each topic (e.g., the number of publications related to a specific topic) across different years as a time series data, and further predict the topic's unseen future score by using time-series forecasting. However, the complication of topic forecasting is that research topics are inherently correlated to each other. Some topics many be strongly correlated, such as social networks and graph mining, whereas other topics may be inversely correlated or have no correlation at all.

Intuitively, one can use multi-variant time series forecasting to predict the popularity of a topic. Nevertheless, the evolution of topics in text has a *low resolution yet high sensitivity* property, which means that a topic's popularity does not change much in days or months (i.e., low resolution), but for a number of consecutive years they may change dramatically and fade quickly (i.e., high sensitivity). As a result, the rising and falling of the topics do not have a long term spanning, which is typically required for traditional time-series forecasting. For example, our experiments in Table 1 show that, out of 54 topics discovered from the data, some emerging topics, such as "social network", have a clear increasing trend, whereas traditional topics, such as "decision trees" or "information retrieval", remain stable and do not have clear trends.

The above observations motivate our research to discover meaningful topics from documents, and further build models to accurately forecast future trend of the topics. In order to identify topics, we propose the use of sentence-level association rule mining to extract strongly correlated (co-occurring) keywords as patterns. For topic trend prediction, we propose a combined random under-sampling of topics framework to build an ensemble of forecasters for time series forecasting in topic prediction (identify a future trend of the pattern). The goal of the combined sampling technique is to improve the forecasting accuracy of the topic forecasting model's performance on scholarly journals conferences (there has been significant progress in the area of topic forecasting in time-sensitive domains such as twitter but very few works exist in scientific publications [10]). Finally, we analyze the relationship between topics and use the clique percolation method (CPM) [11] to find communities of topics and visually demonstrate their relationships.

Our research brings several noticeable contributions and major findings to the areas:

- We propose a fine-grained topic discovery approach using sentence-level pattern mining to discover meaningful topics. Our method is highly scalable and efficient for large scale documents.

Table 1 The 54 topics and their detailed temporal frequency

Regress_logist	2	2	10	11	8	11	12	8	5
Random_walk	0	0	4	4	4	10	9	11	19
Time_seri	45	13	10	38	29	29	36	42	37
Neural_network	14	4	2	2	2	5	7	12	7
Social_network	1	6	6	15	19	32	41	62	72
Compon_princip	2	3	9	5	12	8	11	7	9
Mixtur_model	5	8	17	19	14	22	22	6	12
Detect_anomali	3	9	4	13	6	3	15	26	10
Search_engin	6	1	3	8	9	9	15	18	12
Model_graphic	0	1	11	6	10	14	7	17	7
Itemset_frequent	12	14	28	30	21	16	17	15	13
Transfer_learn	0	0	0	0	1	6	10	19	18
Bay_naiv	7	6	17	26	6	4	9	7	4
Year_recent	4	3	4	4	13	8	9	13	11
Model_probabilist	4	7	18	18	26	22	14	15	33
Neighbor_nearest	3	9	5	4	8	13	6	8	7
Analysi_princip	1	1	7	3	10	7	9	7	8
Page_web	21	8	4	9	2	6	7	16	11
Learn_semi-supervis	1	3	7	11	14	20	22	24	30
Squar_least	0	0	4	3	4	7	13	12	10
Collabor_filter	0	3	10	9	2	8	10	16	20
Model_latent	0	1	3	14	13	15	16	19	24
Graph_edg	1	0	8	6	3	4	5	10	22
Detect_outlier	5	2	10	10	25	8	9	12	19
Process_gaussian	3	0	5	13	2	14	13	23	15
Special_case	1	3	6	8	7	10	6	8	8
Select_featur	18	9	28	35	17	32	47	35	77
Model_markov	5	5	19	20	13	10	19	16	19
Field_random	0	0	16	10	8	16	5	18	17
Inform_retriev	11	4	8	10	10	11	12	17	6
Subgraph_graph	1	4	10	0	7	6	8	4	13
Cluster_spectral	0	0	8	6	7	13	9	9	17
Prefer_user	8	3	3	5	4	6	5	10	10
Scale_larg	4	0	4	11	8	15	18	21	26
Learn_activ	6	3	10	8	15	20	29	28	26
Learn_reinforc	2	0	9	5	9	7	9	7	6
Gene_express	5	13	14	9	12	2	9	5	5
Topic_model	4	0	5	0	22	27	22	56	36
System_recommend	4	4	4	10	8	13	8	10	21
Dimension_high	11	11	15	17	18	7	20	14	26
Use_wide	3	4	8	10	11	7	11	9	18
Pattern_sequenti	12	2	5	15	5	10	13	2	8
Loss_function	1	3	7	7	5	6	18	14	7
Model_gaussian	4	2	4	7	11	10	11	11	8
Model_infer	0	0	9	11	11	9	22	31	21
Optim_convex	0	0	2	3	11	8	5	15	9
Global_local	3	1	7	8	6	2	13	15	14
Associ_rule	42	48	14	21	25	4	20	9	5
Text_categor	9	0	9	2	6	10	8	4	10
Model_build	3	1	5	9	5	7	8	12	8
Tree_decis	12	25	37	12	19	11	11	9	22

Table 1 continued

Dimension_reduct	5	3	6	10	26	19	15	5	22
Machin_support_vector_svm	5	7	12	4	8	9	9	6	7
Pattern_mine_frequent	6	5	10	9	13	10	8	4	12

Each row denotes one topic. The first column denotes the topic, and the 2nd to the 10th columns record the frequency of the topics occurring in the papers published in year 2002, 2003, ... , 2010, respectively

- We propose to employ ensemble forecasting to predict future trends of research topics. Our empirical validation demonstrates strong dependency between topic correlations and the forecasting quality. By utilizing the topic dependency, our ensemble forecasting model achieves good performance gain.
- Our model visually demonstrates the community of research topics, which help understand evolutionary relationships of different topics.

The remainder of the paper is structure as follows. "Related work" section reviews existing work in topic discovery from documents. "Topic discovery and future trend prediction framework" section introduces the proposed topic mining and ensemble forecasting framework. Experiments and results are reported in "Experiments" section, and we conclude the paper in "Conclusion" section.

Related work

As the sheer size of unstructured text data keeps increasing, there is a need to understand and extract information from large scale texts or the world wide web. Modern search engines initially provided two retrieval mechanisms: one allows users to browse documents based on subjects made by humans and the other retrieves documents based on a word query such as a current news topic, a sports topic, etc [12]. For the former, each browsing category requires a cluster label, classification is then used to assign the documents to the browsing category. Examples of using predefined topics to classify documents are wordnet [13] and semantic networks [14] for texts. Earlier approaches to document retrieval based on a query include keyword matching and vector based representations of the word regarding the incidence of words in documents [15], which is also known as information retrieval.

Document clustering

Document clustering is concerned with trying to extract data distribution information from textual unstructured data where there is no prior knowledge [16]. Given a set of documents, clustering uses a distance based approach to separate documents into groups, by using general methodologies such as partitioning and hierarchical clustering [17]. Two modes used to find hierarchical clusters include agglomerative and divisive modes. The divisive mode first assumes that all the data points are part of one cluster as opposed to the agglomerative mode which first assumes that all data points are their own individual clusters. Further details regarding both modes can be found in [17]. For partitioning based clustering algorithms, all documents are partitioned into different groups at the same time by using certain membership criteria, such as euclidean distance measure or probabilistic distribution functions.

Topic discovery

As more and more data became available in the field, simple document clustering became inadequate, because a cluster of documents do not immediately unfold the high level semantics, such as the main theme of the underlying text corpus. As a result, the need to automatically discovery topics from a set of documents becomes necessary.

Topic discovery, also called "topic identification", "topic finding", "label identification", "cluster labeling", or "category labeling" to name a few, aims to find common themes of a set of documents which carry consistent semantic meaning [18].

Sahami [19] proposed a topical space navigation method by using hierarchical clustering to automatically group related documents into sub categories triggered by a user's query. The authors argued that given the rapid growth of the Internet, the topical guides that were previously selected by hand were no longer a possibility. The resulting document groups often resemble a topic category structure automatically imposed on only the set of documents matching a user's query.

Instead of using general predefined concepts to label each cluster, Jayabharathy et al. [20] improved the idea of clustering documents to find topics by using the terms the cluster contains. In addition to partitioning clustering discussed thus far, the hierarchical clustering approach has been widely used for topic discovery as well [17].

While many methods rely on a single clustering methodology for topic discovery, Ayad et al. [21] proposed an aggregation of clustering results to cluster the data. More recently, instead of using the vector approach to mine only the content of twitter messages for topics, a new approach [22] is proposed to mine structures of the graph generated by the interactions between twitter users.

Others [23] have also found practical applications to identifying topics by organizing search results with respect to different topics. However the problem of identifying representative words for the collection of documents still remains a challenging task.

In the following subsection, we briefly discuss some well known methods for topic discovery such as, association rules, latent semantics, and latent Dirichlet allocation. Many methods, such as latent semantics and latent Dirichlet allocation, were originally created for information retrieval.

Association rule based approaches

An association rule has the form $X \rightarrow Y$ where X is a set of antecedent items and Y is the consequent item. The goal is to discover important association rules within a corpus such that the presence of a set of terms in an article implies the presence of another term. An association based topic discovery method was proposed in [24] which represents each document in a vector format, where each document vector is considered a transaction. The association rule mining algorithms can then be applied to the transaction set to extract patterns from the corpus.

In this paper, we consider an association rule a topic candidate. For example, whenever the terms: "machine", "support", and , "vector" are discovered as strongly correlated keywords, either as "support vector machine" or "support vector", it is highly probably that these patterns are related to one topic, i.e., "SVM". Therefore, we will use association rules as candidates, and redefine the patterns to discovery topics.

Latent semantics approaches

Latent semantics approach allows one to compute whether two documents are topically similar, even if the two documents do not have any words in common.

In 1988 Scott Deerwester along with his team obtained a patent for latent semantic analysis (LSA). The general idea for the analysis is to create a low dimensional space of terms. The new space is referred to as "latent" because it does not directly corresponds to any single terms, but represents a combination of the terms in the original input space. As a result, LSA can be used to discover semantic similarity between documents [25]. This procedure addresses the problem of polysemy and synonymy [26].

LSA can be used to find similarity between topics since it has been shown to mimic word sorting and category judgement by humans [27].

Probabilistic latent semantics approaches

Probabilistic latent semantics analysis (pLSA) added a sound probabilistic model to LSA. It is related to non negative matrix factorization [28].

pLSA was introduced in 1999 by Thomas Hofmann as a way to provide a sound statistical foundation [29] to LSA. In 2006 Newman and Block proposed to use pLSA for topic discovery for finding topics using historical data [7].

Latent Dirichelt allocation approaches

Latent Dirichelt allocation is similar to pLSA except that in LDA the topic distribution is assumed to have a Dirichlet prior. In practice, this results in a more acceptable combination of topics in a document [8]. In particular, this method is from the family of Bayesian nonparametric approach [17], which was developed to improve the mixture models. There have been other variations, such as [30], which can be applied to textual data and do not require extensive corpus engineering.

In 2003, David Blei, Andrew Ng and Michael Jordan proposed Latent Dirichlet allocation model (LDA) to address some of the shortcomings of pLSA [31].

Zhu et al. [32] used LDA by defining topics based on a single word to find topics in people such as their activities in social media. All of these approaches employ an unsupervised style of learning to cluster the data where all the documents are represented in the vector space, it is in this manner that clusters are then assumed to be topics.

Topic discovery and future trend prediction framework

Overall framework

Our proposed topic discovery and forecasting framework is shown in Fig. 1. Overall, the framework includes the following major steps [33]:

1. *Document to transaction transformation* Given a text corpus which contains documents collected from a number of years, each text document is first preprocessed to form transactions.
2. *Topic discovery from transactions* Association rule mining is applied to the transaction set to discover patterns as research topics.

3 *Temporal topic feature characterization* Validating the frequency of each research topic in the text corpus and generate a feature vector for each topic.
4 *Topic correlation* Calculating correlations between discovered topics.
5 *Topic community discovery* Using topic correlations to find research topic communities.
6 *Ensemble topic forecasting* Using ensemble forecasting model to predict popularity of each research topic.

In order to transform a text document into suitable format for pattern mining, we first convert each sentence of the document to form a bag-of-word transaction. By doing so, each sentence is represented as a vector where each dimension corresponds to a keyword and its value is a binary indicator denoting whether the keyword occurs or not (In this paper, keyword and word are interchangeable terms).

In Fig. 1, the collection of the word vectors is named the binary incidence matrix. After the collection of the matrix, we carry out frequent pattern mining from the incidence matrix to discover association rules in the form of $X \rightarrow Y$, where X is a set of antecedent items and Y is the consequent item. We consider each resulting association rule as a set, where each member of antecedent items in X and the item in Y are members of the set.

On the third step (step 3), we build the topic incidence matrix for documents collected for each year, by using both the topics and binary incidence plus year data set to generate each vector (see Fig. 1). For each topic, the vector will record the number of times the topic appear in each year's papers. As a result, the vector will record temporal evolution information of the topics, through which our forecasting module can predict the future trend of each topic.

Association analysis for topic discovery

In order to mine association rules as patterns, we download research papers published in selected conference proceedings (detailed in "Benchmark data" section). We use the title and the abstract of each paper for processing, and then apply the following natural language processing techniques [34]:

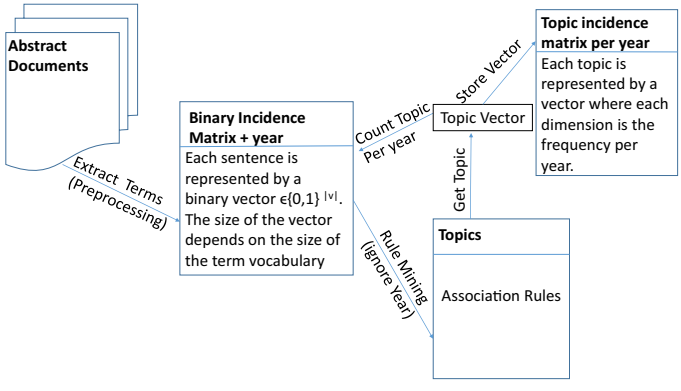

Fig. 1 The flowchart of the proposed framework. Detailed information about topic discovery is explained in "Topic discovery and future trend prediction framework" section

1 Tokenization and case conversion.
2 Removing stops words and small words.
3 Part of speech tagger (POS).
4 Removing verbs.
5 Stemming (reduce to word stem).
6 Lemmatization.

Figure 2 shows an example of the application of the above mentioned language processing techniques to a given abstract. Because our processing unit is a sentence, we have to analyze the document at sentence and word level. Therefore, we apply tokenization to both the sentences and words. For each sentence, we tokenize each word and apply all the subsequent natural language processing techniques one sentence at a time. Our goal is to make sure that each word is truly representative for each sentence. It is important to mention that we treat the title of each document as a sentence, because title provides highly summarized description of the whole document. After the stop words and small words removal steps, words such as "between" and "and" will be removed and the entire corpus is transformed to lowercase. We also remove stop words, such as "This", "on", and small words, because these words are considered common words that are highly pervasive in every sentence.

After the above process, we employ a POS [35] to label all the words with the corresponding part of speech label. In this manner, we would remove verbs such as "evaluates", "assessed", "propose" etc., which are very common in scientific publications. During the verb removal step, verbs such as "exploiting" and "adapting" were removed. Verbs are helpful in describing actions but are not themselves topics and as such we remove the verbs. In addition, we also remove all different variations of the same word (stemming). Once we take all of the steps above, the data is considered to be normalized.

Finally, the last two steps are similar since they both attempt to remove the inflectional variations of the words by using two different approaches, one approach just chops the word using heuristics (stemming) and the other approach uses a dictionary (lemmatization) in order to remove the inflections.

In the end by applying natural language processing techniques, we accomplish two things. First, we remove unnecessary words such as stop words and verbs. Second, we reduce the ambiguity of the remaining words by converting each word to lower case, inflection free versions of themselves for all sentences. Our main objective during this prepossessing phase is to transform each sentence to a transaction of bag of words. Figure 3 depicts the final binary transformation from a raw text to binary representation of a bag of words.

In Fig. 3, we demonstrate an example of transforming an abstract into a vector space representation of each sentence, by using the above normalization process. The vector space, as shown in Fig. 1, is called the matrix of the binary incidence matrix. Each transaction is a representation of the incidence of a word from a vocabulary, the vocabulary or bag of words is the set of words after all the steps to normalize the data have been taken. Thus, using this vocabulary we generate a binary incidence matrix as show in Fig. 3, which shows the prevalence of ("document", "cross-domain", "text") given that it appears in two out of four sentences. For this particular example, it is easy to find that

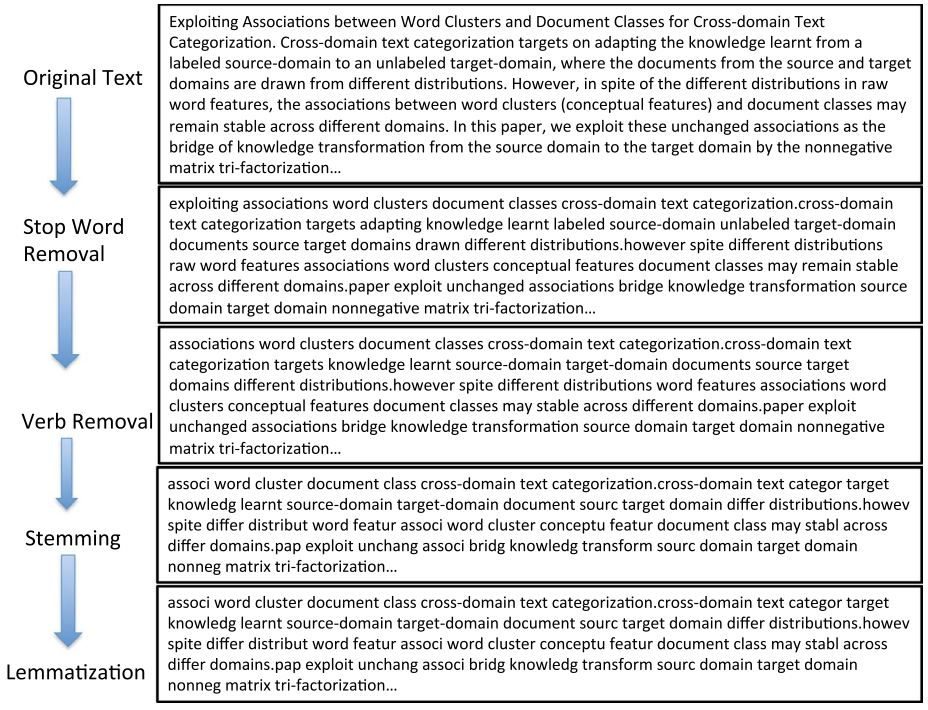

Fig. 2 The flowchart of preprocessing. Detailed information about the normalization process

"document", "cross-domain" and "text" frequently appear as a pattern, so we can consider "document_cross-domain_text" as a topic, if we can automatically find such patterns.

Pattern mining and refinement

Even from a small example we can appreciate that the data is sparse, where each transactions t_i only have very few words. So we use frequent itemset mining algorithm in WEKA [36] to discover association rules from the transaction set.

Because multiple association rules may contain redundant words, we cannot simply treat each discovered association rule as a topic. For example, "association mining" and "association rule mining" are two association rules (or patterns), but they really belong to only one topic. So we carry out a rule refinement process by considering each topic set as an object in a larger set called the set of association rules. We use the definition of sets to allow multiple relevant association rules to absorb and merge as one topic. In our process, we eliminate all the proper subsets from being considered topics, if an object from the association rules is a proper subset of another object then we delete this object. As a result, each final set will be regarded as a topic. By doing so, we can ensure different ordering of the words always results in one unique topic and therefore avoid topic duplication.

In order to forecast future trend of each topic, we iterate through the list of discovered topics and refer back to the binary incidence matrix and year dataset to obtain each dimension for a topic vector (as shown in Fig. 1). Thus using the example provided in Fig. 3 and assuming that the data provided is the entire data for a full year. The resulting vector for the pattern "document_cross-domain_text" will be $v = \langle 2 \rangle$ because the count

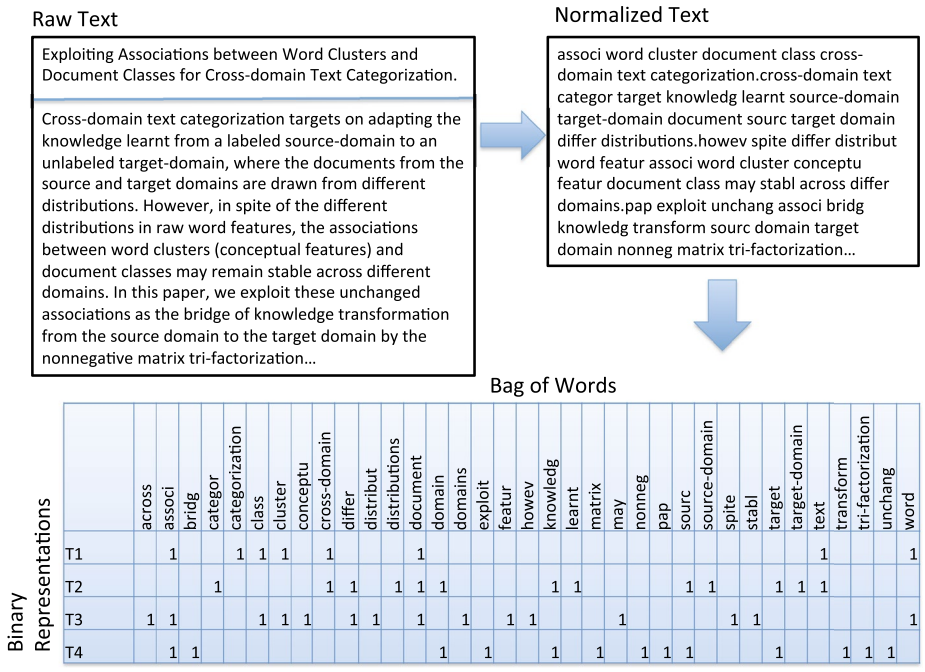

Fig. 3 Vector space representation. Each document is first normalized, and then converted into a binary transaction format

for items "document", "cross-domain" and "text" is 2. We store all the topic vectors along with their dimensions in a data file called the topic incidence matrix per year. We use the topic incidence matrix for topic correlation analysis and later for time series forecasting.

Temporal topic correlation analysis

The topic vectors generated in the previous subsection allow us to use correlation coefficient to generate a data set of correlations among all topics.

The correlation co-efficient provides the measure of the strength of two random variables (vectors). Specifically, Pearson's correlation is provided as follows [12]. Assume X and Y are two variables with a set of observed values, such that $X = \langle x_1, x_2, \ldots, x_n \rangle$ and $Y = \langle y_1, y_2, \ldots, y_n \rangle$.

Correlation coefficient between two random variables X and Y is defined as

$$\rho(X, Y) = \frac{\mathbf{Cov}(X, Y)}{\sqrt{\mathbf{Var}(X)\mathbf{Var}(Y)}} \tag{1}$$

where the *sample covariance* between X and Y is defined as

$$\mathbf{Cov}(\mathbf{X}, \mathbf{Y}) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n - 1} \tag{2}$$

The *sample variance* of random variable X is defined as

$$\mathbf{Var}(X) = \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n - 1} \tag{3}$$

Thus the *correlation coefficient* between two random can be rewritten as

$$\rho(X,Y) = \frac{\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{n-1}}{\sqrt{\frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{n-1} \times \frac{\sum_{i=1}^{n-1}(y_i-\bar{y})^2}{n-1}}} \tag{4}$$

The formula can be further simplified to

$$\rho(X,Y) = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \times \sum_{i=1}^{n-1}(y_i-\bar{y})^2}} \tag{5}$$

The correlation co-efficient value ranges between $+1$ and -1, with 1 denoting the strongest correlation between two random variables, and a negative value denoting inversely correlated variables.

Example If we look at the two topics reported in Table 2, and the the two topic vectors are represented by X and Y.

$$X = [x_1, x_2, \ldots, x_5] = [14, 4, 2, 2, 2]$$

$$Y = [y_1, y_2, \ldots, y_5] = [1, 6, 6, 15, 19]$$

Their means are calculated as follows:

$$\bar{x} = \frac{14+4+2+2+2}{5} = 4.8$$

$$\bar{y} = \frac{1+6+6+15+19}{5} = 9.4$$

Using formula 1, the correlation between X and Y is calculated as follows: $= \frac{[((14-4.8)\times(1-9.4))+((4-4.8)\times(6-9.4))+((2-4.8)\times(6-9.4))+((2-4.8)\times(15-9.4))+((2-4.8)\times(19-9.4))]}{\sqrt{((14-4.8)^2+(4-4.8)^2+(2-4.8)^2+(2-4.8)^2+(2-4.8)^2)\times((1-9.4)^2+(6-9.4)^2+(6-9.4)^2+(15-9.4)^2+(19-9.4)^2)}}$

$$\rho(X,Y) = -0.6999514$$

which concludes a moderate inverse correlation between two topics (i.e., the decrease of X implies the increase of Y).

Topic community discovery

In order to find a set of topics which are strongly correlated to each other (or strongly inversely correlated topics), we use correlation co-efficient between two topics to build

Table 2 The two topic examples and their detailed temporal frequency

Topic	Variable	2002	2003	2004	2005	2006
Neural_network	X	14	4	2	2	2
Social_network	Y	1	6	6	15	19

a graph. Each node of the graph denotes a topic, and an edge between nodes denotes degree of correlation between two topics. This graph structured network allows us to explicitly model interactions between topics.

In order to discover set of strongly correlated topics, we set a cutoff threshold value to remove edges whose value is less than a threshold. After that, we use a CPM based community detection algorithm (which was implemented in CFinder [11]) to find group of topics with strong correlations.

CPM is a method to find overlapping communities [37], where nodes inside a community are strongly correlated to each other, compared to their correlation with nodes outside the community. CPM first identifies all cliques with size k (where k is a user specified parameter) from the network. Using the definition of clique adjacency, CPM regards two cliques as being adjacent if they share $k - 1$ common nodes. We then form an adjacency graph where the names of each node are now the combined names of all the nodes in a clique, and an edge exists only if two clique nodes are adjacent. The adjacent cliques are then connected to form a node group (i.e., a community). In other words, each fully connected subgraph in the adjacency network (graph) is considered a community.

An example of the discovered topic community using CPM is show in Fig. 11.

Ensemble forecasting for topic trend prediction

In order to forecast the future trend of a topic, we employ the WEKA forecasting plugin [38], which provides an environment for time series analysis and a forecasting model. WEKA's forecasting tool employs a set of regression based learning methods to model and predict time series data. This is achieved by using a special encoding approach to add additional input fields to represent time dependency in the original data. So the original time series data can be directly converted into a generic format that "standard propositional learning algorithms can process" [38]. An inherent advantage of such transformation is that a forecasting problem can be immediately solved by using any regression models in the WEKA tool. For example, a recent research employed the WEKA forecasting tool to successfully predict polarity trend of public sentiment on YouTube [39]. Using a methodology that "aggregated the sentiment for comments on a weekly basis", by calculating the average sentiment for each week. This allowed them to forecast sentiments for the future 26 weeks.

In order to forecast one specific topic, we need to select a target field (or target topic), which is the field of the data needs to be forecasted. We can select one target field (i.e., selecting one topic as the target). In this case, the forecasting model will only use the time series historical data of the selected field to predict the future value of the same field. In addition, we can also select multiple fields as targets, then the system will use time series of multiple topics to predict their future values.

A potential advantage of selecting multiple fields as targets is that, compared to forecasting using a single field, temporal correlations of different topics may help improve the forecasting accuracy. For example, if "social networks" and "graph mining" are strongly correlated, we may select both of them as target fields, and use their temporal correlations to help improve the forecasting accuracy. Accordingly, the fields to be selected as the targets will affect the final forecast for any given target topic. Due to this

reason, we need to decide that when forecasting a future value for a topic X_i, which topic(s) should also be selected to help forecast X_i's future value.

Our approach is to employ an ensemble of forecasters and aggregate their combined forecasting power to provide a better forecast for X_i (as shown in Fig. 4). For any given forecaster that forecasts for topic X_i, we set the target selection to topic X_i along with N randomly chosen fields, excluding X_i. Additionally, we also generate $M-1$ forecasters for X_i. We aggregate their forecast values by giving each forecaster equal weight. As a result, the final forecast is equal to Forecast 1 + Forecast 2 + Forecast 3 + $\cdots$ + Forecast M divided by M. Our objective is to see how many randomly generated fields (N) yields the best forecast. The effectiveness of each forecaster is evaluated by R squared value, *MSE* (mean squared error), and the average standard deviation of its error, which will be soon be demonstrated int the experiments.

Experiments

Benchmark data

The data used for experimental study was collected from scientific papers published in four premier data mining and machine learning conference forums, including ACM-KDD, IEEE-ICDM, SIAM-SDM, and ICML conferences. We use both the abstract and

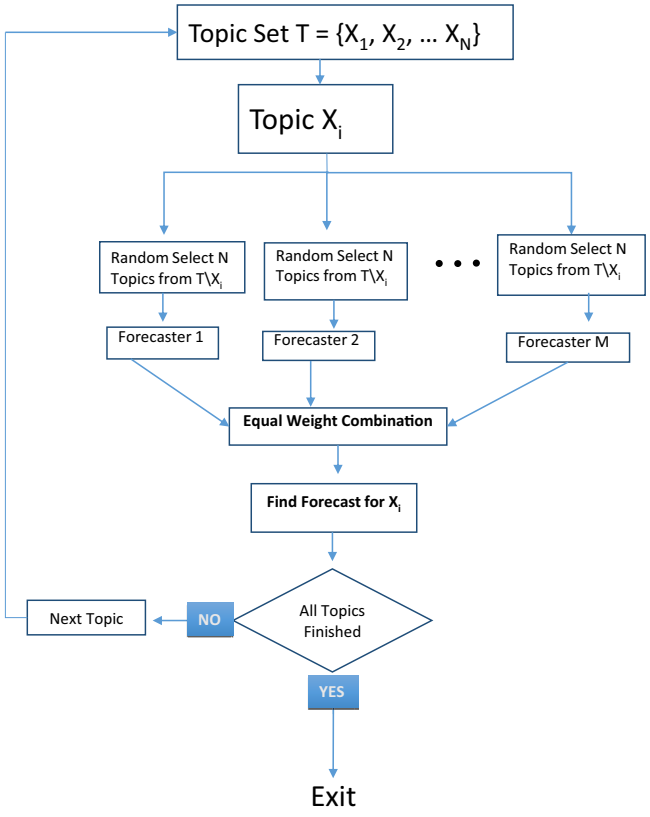

Fig. 4 The proposed ensemble forecasting framework (detailed in "Topic discovery and future trend prediction framework" section)

title of each paper. The years that were taking into considerations are: 2002–2010, which result in 6122 papers in our dataset over the 9 year period.

The majority of the data used for this experiment was collected by Tang et al. [40]. This dataset is mainly geared for the study of co-authorship networks (social networks), thus a considerable number of papers do not have abstracts. In order to compensate for this unavailability of data, we either crawled the data from the conferences websites or used web service to retrieve the abstracts, if this service was available. The downloading of the data, normalization of the data, and the sentence to transaction translation was conducted by python script. The output from this script generated a .arff file for WEKA.

Two java programs with the WEKA API were used. One program intends to mine association topics and the other one aims for forecasting using the forecast plugin. In order to reduce the dimensionality of the topic incidence file, we further took steps to only include words appeared more than twice in the entire database for the association mining. We used the Appriori model from WEKA for the association mining. For the underlying classifiers for forecasting, we used linear regression from WEKA.

Topic discovery results

For the topic discovery experiment, we changed the minimum support values and the confidence for frequent pattern mining, which results in two set of topics (two topic datasets). The first set contains 54 topics, and the second set contains 133 topics. The set of 54 topics (along with their frequency in each year) is reported in Table 1. The results show that out of the 54 discovered topics, only four of them (marked in italic format) are not meaningful topics and the remaining 50 are legitimate topics. This results in a 92.3 % precision rate for topic discovery.

Topic forecast benchmark methods

In order to forecast future trend of the topic, we use the following benchmark methods: Using All, Highest Correlated field, Highest Correlated field plus one. All the benchmark methods jointly model multiple target fields simultaneously and capture different dependencies between the topics we choose. Using All generated a forecast by using all topics to help forecast for target topic (54 for 54 topics, and 133 for 133 topics). This methods assumes that all topics are dependent to each other. We also used the highest correlated field along with the chosen topic as a benchmark method as well. This method assumes that only the highest correlated topic is necessary for the successful prediction of a topic. We also used the highest correlated field along with one randomly generated field along with the chosen topic as well. This method also uses a ensemble forecasting by generating 100 different forecasting where each forecaster varies by which topic was randomly chosen for target selection. The final forecast is the average of all forecasters.

We analyzed 6122 papers from the conferences mentioned above.

Measures

We first provide SSE, SSY, and SSO since the rest of the formulas are based on this formulas.

$$SSE = \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \qquad SSY = \sum_{i=1}^{n} {y_i}^2 \tag{6}$$

$$SSO = n\bar{y}^2 \qquad SST = SSY - SSO \tag{7}$$

$$SSR = SST - SSE \qquad R\text{-}squared = \sqrt{\frac{SSR}{SST}} \tag{8}$$

R-squared value is a statistical measure evaluating the closeness of the data to the fitted regression line. It is also being referred to as the coefficient of determination, or the coefficient of multiple determination for multiple regression.

A zero value indicates that the model explains none of the variability, the larger the *R* value, the better the data fits to the fitted regression line (the largest *R* value is 1).

$$MSE = \frac{SSE}{n} \qquad SDE = \frac{\sum_{i=1}^{n}(Err_i - \bar{E}rr_i)^2}{n} \tag{9}$$

Standard deviation of the error (SDE): For all the forecaster we compute the error for each field in the prediction, and calculate the standard deviation for each forecaster. The final results with respect to each performance metric are reported in Figs. 5, 6, 7, 8, 9 and 10. The comparisons of the topic forecasting results, with respect to all 54 topics, are reported in Table 3.

Topic forecasting results

We first discuss the results of the 54 topics. Based on the coefficient of determination (*R*-squared), the best results were achieved when using 6 and 4 randomly generated fields over 100 runs. The third best value according to *R* value was using 7 random fields. However, when looking at their corresponding error, 7 is the worst offender, followed by 6 and finally 4. By looking at their MSE, 6 and 4 are the best, therefore we choose these two forecasters.

Now we discuss the results of the 133 topics experiment. Based on the coefficient of determination (*R*-squared), the best results were achieved when using 7 and 9 randomly generated fields over 100 runs. The third best value, according to *R*-squared value, was using 8 random fields. When looking at their corresponding error, 7 is the worst offender, followed by 8 and finally 9.

The experiments in Fig. 7 show that, compared to the forecasting using a single target fields, using a very few number of target fields, such as 2–7 can almost always help improve the forecasting accuracy, as shown in the left corner of Fig. 7. This is indeed consistent with our hypothesis that using multiple target fields can help improve the forecasting results. Moreover, when continuously adding more target variables, as shown

Table 3 Topics forecasting result comparisons (54 topics)

	Using all	Highest correlated field	Highest correlated field +1 random	Six random	Four random
R	0.6260	0.7081	0.7144	0.7325	0.7123
MSE	118.2200	96.9170	95.1681	90.0888	96.3750
Average_std	NA	NA	4.3078	7.9458	7.3928

Each row denotes a performance metric, and each row denotes a forecasting method. "Six random" means using six randomly selected topics and the target topic for forecasting

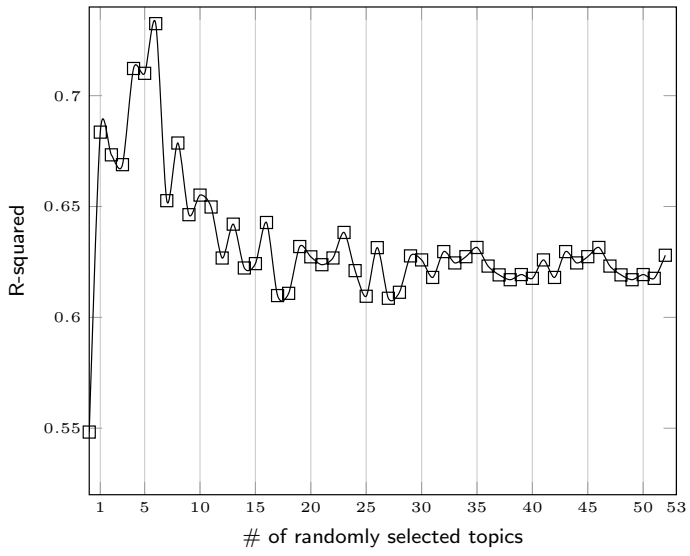

Fig. 5 *R*-squared values of forecasters using a random number of topics (54 topics)

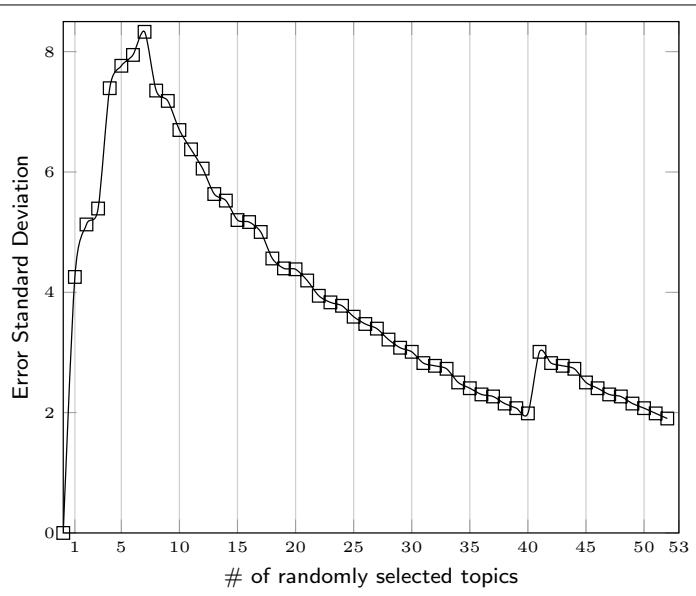

Fig. 6 Error standard deviation of forecaster using a random number of topics (54 topics)

on the right section of Fig. 7, the forecasting accuracy will actually drop, which suggests that too many correlations may add confusion to the forecasting modules and result in deteriorated results (similar to the case in supervised learning where continuously adding extra features may in fact deteriorate the classification accuracy).

Based on the above results we find that using 7 and 9 randomly selected fields generate the best performance, so we choose these two forecasters in the experiments.

We compare the two best results from our ensemble method with the benchmark methods (we first report the 54 topic experimental results, and followed by the 133 topics experiment). We notice that all the methods are better than using all the columns

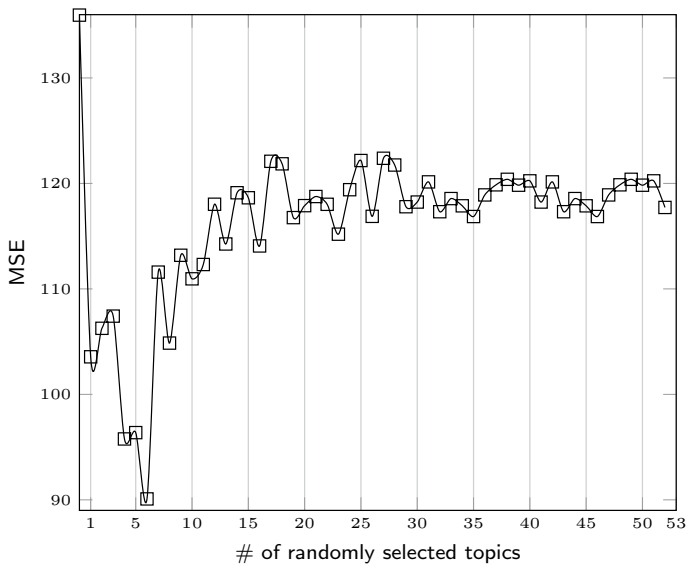

Fig. 7 MSE values of forecasters using a random number of topics (54 topics)

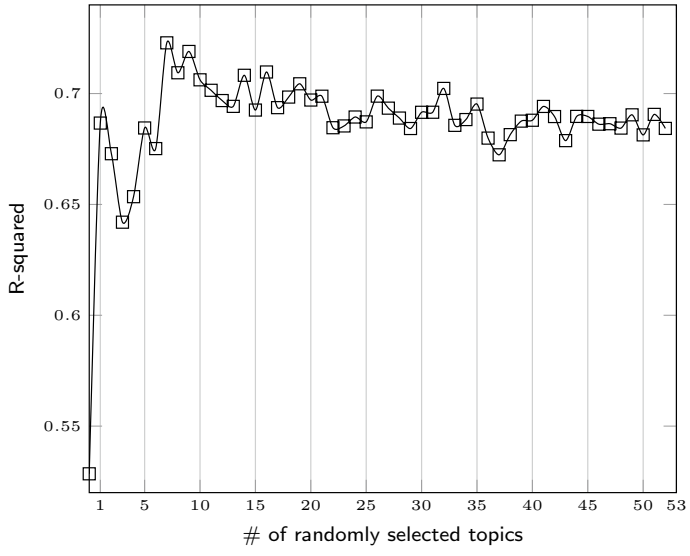

Fig. 8 *R*-squared values of forecasters using a random number of topics (133 topics)

forecasters. The best method is the one that uses six random topics to help forecast a target topic. Please note that there is no average standard deviation for both "Using All" and "Highest correlated" since there is no variation in these methods.

For the 133 topics, we notice that all the methods are also better than using all the highest correlated forecaster. The best method is the one that uses 7 random fields. Intuitively, when selecting multiple target fields for forecasting, it makes sense to use strongly correlated fields to build forecasters. Unfortunately, our experiments show that using strongly correlated fields as target is, in fact, not always helpful. This is possibly because strongly correlated fields will introduce strong correlations in transformed

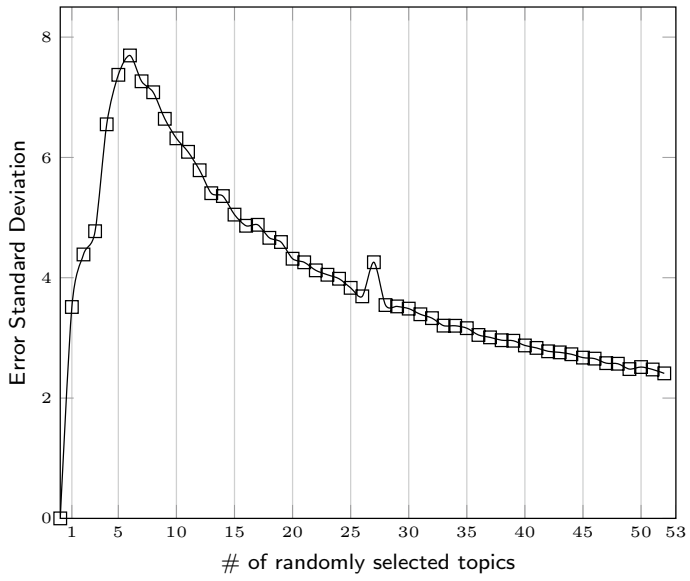

Fig. 9 Error standard deviation of forecasters using a random number of topics (133 topics)

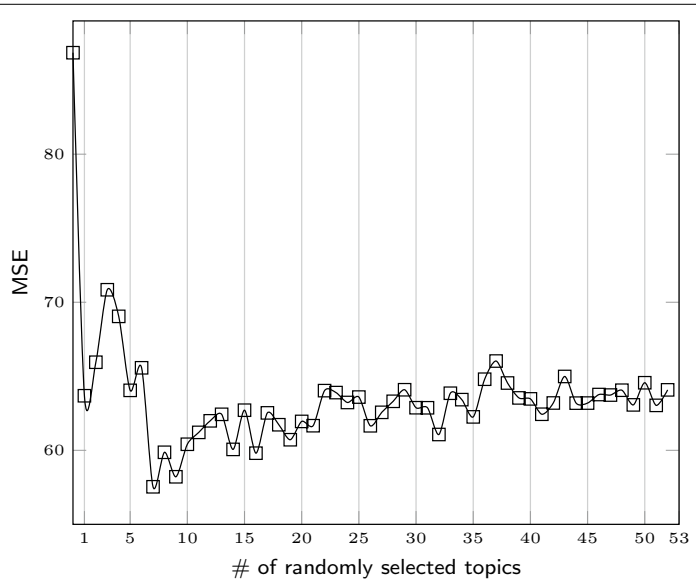

Fig. 10 MSE values of forecasters using a random number of topics (133 topics)

dataset (as we have already explained about the WEKA data transformation process for forecasting [38]). It is well known that feature correlations will introduce complication in supervised learning, and possibly deteriorate the learning results. In comparison, using randomly selected target fields achieve the best performance.

As we have explained, there is no average standard deviation for both "Using All" and "Highest correlated" since there is no variation in these methods.

Topic community discovery results

In Figs. 11 and 12, we report the topic network and communities discovered from 54 topics and 133 topics, respectively. In each figure, the node denotes a topic and the edge connects two topics (we removed the edges whose correlation values are less than a threshold). Each shaded region denotes a community. The results show that research topics do form strongly correlated groups, and many topics co-evolve together.

Conclusion

In this paper we proposed to discover meaningful topics from documents and forecast their future trend. To discover topics, we converted each sentence of the document into a transaction format, and employed association rule mining algorithms to discover frequent patterns from the documents. By using set inclusion/exclusion operations, we refined the frequent patterns as topics and further recorded their temporal frequency, across a number of years, to characterize temporal evolution of each topic. By using correlation coefficient to discover correlations between topics, we are able to discover a set

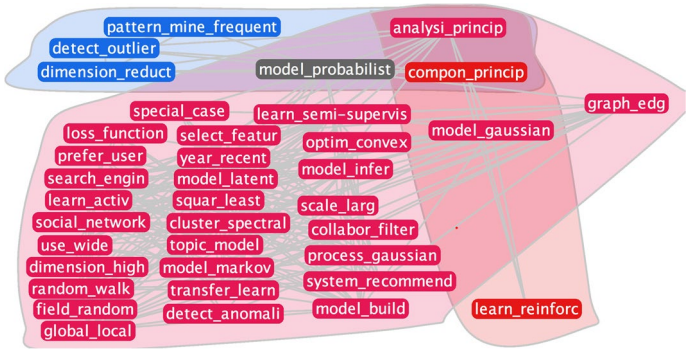

Fig. 11 Topic network and community. Each node denotes a topic and an edge indicates a correlation greater than the threshold. Each shaded region denotes a community (*i.e.* strongly correlated topics) (54 topics)

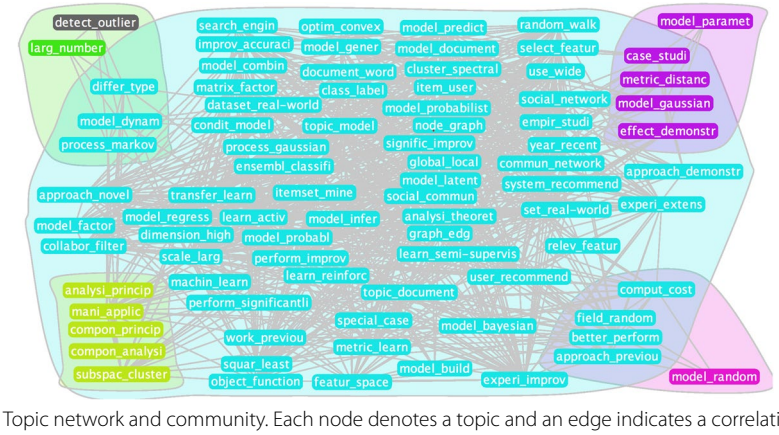

Fig. 12 Topic network and community. Each node denotes a topic and an edge indicates a correlation greater than the threshold. Each shaded region denotes a community (*i.e.* strongly correlated topics) (133 topics)

of strongly correlated topics as a community. In addition, we also proposed an ensemble forecasting model to predict the future trend of a topic.

Our experiments and validations from papers published in four world premier data mining and machine learning conferences (including 6122 papers) confirmed that the proposed combined prediction framework yields better performance, with respect to the R-squared values, MSE and the standard deviation of the error, than baseline methods.

Authors' contributions

JH carried out the literature review and performed the experiments, analysis, and wrote the draft of the manuscript. AA helped revised the submission and provided comments for topic detection. XZ wrote the abstract and helped organize the structure of the manuscript, he also introduced the research topic to JH and coordinated with the authors to complete and finalize this work. All authors read and approved the final manuscript.

Acknowledgements

This research is partially supported by US National Science Foundation under Grant No. IIP-1444949. A preliminary version of the paper [1], 4 pages, was published in the 16th IEEE International Conference on Information Reuse and Integration, San Francisco, CA, August 13–15, 2015.

Competing interests

The authors declare that they have no competing interests.

Received: 13 November 2015 Accepted: 18 February 2016
Published online: 14 April 2016

References

1. Hurtado J, Huang S, Zhu X. Topic discovery and future trend prediction using association analysis and ensemble forecasting. In: the 16th IEEE international conference on information reuse and integration. San Francisco, CA: 2015.
2. Mei Q, Zhai C. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In: Proceedings of the eleventh ACM SIGKDD international conference on knowledge discovery in data mining. 2005.
3. Berlanga-Llavori R, Anaya-Sánchez H, Pons-Porrata A, Jiménez-Ruiz E. Conceptual subtopic identification in the medical domain. In: Geffner H, Prada R, Machado Alexandre I, David N, editors. Advances in artificial intelligence—IBERAMIA 2008. Lecture notes in computer science, vol 5290. Springer; 2008. p. 312–21.
4. Mörchen F, Dejori M, Fradkin D, Etienne J, Wachmann B, Bundschus M. Anticipating annotations and emerging trends in biomedical literature. In: Proc. of ACM SIG KDD conference. 2008.
5. Tucker C, Kim H. Predicting emerging product design trend by mining publicly available customer review data. In: Proc. of international conference on engineering design. 2011.
6. Schumaker R, Chen H. Textual analysis of stock market prediction using breaking financial news: The azfin text system. ACM Trans Inf Syst. 2012;27(2):1–19.
7. Newman DJ, Block S. Probabilistic topic decomposition of an eighteenth-century american newspaper. J Am Soc Inf Sci Technol. 2006;57(6):753–67.
8. Blei DM. Introduction to probabilistic topic models. Commun ACM. 2012;55(4):77–84.
9. Fu T-c. A review on time series data mining. Eng Appl Artif Intell. 2011;24(1):164–81.
10. Wang X, McCallum A. Topics over time: a non-markov continuous-time model of topical trends. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining. 2006.
11. Palla G, Derényi I, Farkas I, T T.V. Uncovering the overlapping community structure of complex networks in nature and society. Nature. 2005;435(7043):814–8.
12. Mettrop W, Nieuwenhuysen P. Internet search engines—fluctuations in document accessibility. J Doc. 2001;57(5):623–51.
13. Liu Y, Scheuermann P, Li X, Zhu X. Using wordnet to disambiguate word senses for text classification. In: international conference on computational science. 2007.
14. Sussna M. Word sense disambiguation for free-text indexing using a massive semantic network. In: Proceedings of the second international conference on information and knowledge management (CIKM). 1993.
15. Wiemer-Hastings P, Wiemer-Hastings K, Graesser A. Latent semantic analysis. Proceedings of the 16th international joint conference on artificial intelligence. 2004. p. 1–14.
16. Joshi AC, Padghan VR, Vyawahare JR, Saner SP. Enforcing document clustering for forensic analysis using weighted matrix method (wmm). 2015.
17. Jain AK. Data clustering: 50 years beyond k-means. Pattern Recognit Lett. 2010;31(8):651–66. Award winning papers from the 19th international conference on pattern recognition (ICPR).
18. Stein B, Eissen SMZ. Topic identification: framework and application. In: Proc of international conference on knowledge management (I-KNOW). 2004.
19. Sahami M. Using machine learning to improve information access. Technical report, Stanford University; 1998.
20. Jayabharathy J, Kanmani S, Parveen AA. Document clustering and topic discovery based on semantic similarity in scientific literature. In: Communication software and networks (ICCSN), 2011 IEEE 3rd international conference on. 2011. p. 425–9.

21. Ayad H, Kamel MS. Topic discovery from text using aggregation of different clustering methods. Proceedings of the 15th conference of the Canadian society for computational studies of intelligence on advances in artificial intelligence., AI 02London, UK, UK: Springer; 2002. p. 161–75.
22. Hromic H, Prangnawarat N, Hulpuş I, Karnstedt M, Hayes C. Graph-based methods for clustering topics of interest in twitter. In: Engineering the web in the big data era. Lecture notes in computer science, vol 9114. Springer; 2015. p. 701–4.
23. Wartena C, Brussee R. Topic detection by clustering keywords. In: Database and expert systems application, 2008. DEXA 08. 19th international workshop on. 2008. p. 54–8.
24. Wong PC, Whitney P, Thomas J. Visualizing association rules for text mining. 1999.
25. Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R. Indexing by latent semantic analysis. J Am Soc Inf Sci. 1990;41(6):391–407.
26. Dumais ST, Furnas GW, Landauer TK, Deerwester S, Harshman R. Using latent semantic analysis to improve access to textual information. Proceedings of the SIGCHI conference on human factors in computing systems., CHI 88 New York, NY, USA: ACM; 1988. p. 281–5.
27. Landauer TK, Foltz PW, Laham D. An introduction to latent semantic analysis. Discourse Process. 1998;25(2–3):259–84.
28. Steyvers M, Griffiths T. Probabilistic topic models. Handb Latent Sem Anal. 2007;427(7):424–40.
29. Hofmann T. Probabilistic latent semantic indexing. Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval, SIGIR 99New York, NY, USA: ACM; 1999. p. 50–7.
30. Yano T, Cohen WW, Smith NA. Predicting response to political blog posts with topic models. Proceedings of human language technologies: the 2009 annual conference of the North American chapter of the association for computational linguistics, NAACL 09Stroudsburg, PA, USA: Association for Computational Linguistics; 2009. p. 477–85.
31. Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. J Mach Learn Res. 2003;3:993–1022.
32. Zhu D, Fukazawa Y, Karapetsas E, Ota J. Intuitive topic discovery by incorporating word-pair connection into lda. Proceedings of the the 2012 IEEE/WIC/ACM international joint conferences on web intelligence and intelligent agent technology-, vol 01. WI-IAT 12Washington, DC, USA: IEEE Computer Society; 2012. p. 303–10.
33. Hurtado J, Taweewitchakreeya N, Zhu X. Who wrote this paper? learning for authorship de-identification using stylometric featuress. In: Information reuse and integration (IRI), 2014 IEEE 15th international conference on. 2014. p. 859–62.
34. Loper E, Bird S. NItk: the natural language toolkit. Proceedings of the ACL-02 workshop on effective tools and methodologies for teaching natural language processing and computational linguistics-, vol 1. ETMTNLP 02Stroudsburg, PA, USA: Association for Computational Linguistics; 2002. p. 63–70.
35. Toutanova K, Manning CD. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In: Proceedings of the joint SIGDAT conference on empirical methods in natural language processing and very large corpora. 2008.
36. Witten IH, Frank E, Hall MA. Data mining: practical machine learning tools and techniques. 3rd ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 2011.
37. Ravaee H, Masoudi-Nejad A, Omidi S, Moeini A. Improved immune genetic algorithm for clustering protein-protein interaction network. In: BioInformatics and bioEngineering (BIBE), 2010 IEEE international conference on. 2010. p. 174–9.
38. Community P. Using the weka forecasting plugin. In: Pentaho BI suite community edition. 2011. http://wiki.pentaho.com/display/DATAMINING/Using+the+Weka+Forecasting+Plugin
39. Amar Krishnay JZ, Krishnan S. Polarity trend analysis of public sentiment on youtube. In: The 19th international conference on management of data (COMAD). 2013.
40. Tang J, Zhang J, Yao L, Li J, Zhang L, Su Z. Arnetminer: extraction and mining of academic social networks. Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining, KDD 08 New York, NY, USA: ACM; 2008. p. 990–8.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Immediate publication on acceptance
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com