A Novel Ontology based Framework for Trend Detection from Research Paper Titles

Swaraj K P, Dr. Manjula D, Adhithyan V, Hemnath K B, Manish Kumar L

Abstract—Classification of research papers based on topics like Data Mining, Software Engineering, and Networks etc. is generally done manually. Nowadays Researchers are particularly interested in knowing trending subtopics and keywords under their area of interest and the relevant research articles published in those areas. In this article, a fast novel approach is proposed to find trending topics and subtopics from large cluster of research paper titles for the past n years. Also, an effective method is presented to retrieve paper titles based on a topic or subtopic from trending topics. Finally a ranking algorithm is suggested based on which the resultant paper titles are ranked before being displayed. In this study, titles of papers for last 3 years from DBLP computer science dataset were extracted and stored in a database. Even though, articles from only last 3 years were considered, this system can be easily scaled to accommodate any size and time period. Experiment studies revealed that the proposed trend detection framework is accurate, efficient and scalable.

Index Terms— DBLP, Hadoop, Map Reduce, Ontology, Owl, Sparql, Trend detection, Topic detection

1 INTRODUCTION

From its initial stages itself, the domain of Computer Science has been extremely wide and consistently evolving contributing much to the industry and societal advancement. This undeniable fact is emphasized by a countless numbers of technological advancements and plethora of exploratory papers which are published consistently scattering over wide range of established and emerging topics in the field of Computer Science and Engineering. Worldwide, numerous research articles are published every year covering wide span of topics in this relevant domain. Currently, there is no well-known system to detect trending topics from research paper titles and also to retrieve important research paper titles from a trending sub-topics.

DBLP is a bibliographic dataset in xml format, hosted at University of Trier, in Germany, which provides metadata about computer science publications. It is now used by many computer scientists for getting an idea about the titles and authors in computer science domain and also for research purpose [6]. This dataset has meta data of more than 2.5 million articles and spans a wide range of computer science topics such as Artificial Intelligence, Databases, Data Mining, Cloud computing, Information Extraction and so forth, which can be further classified into several sub-topics, sub-sub-topics and so on. The xml Meta data is a treasure of rich information and has been enticing considerable research interest from researchers for the last couple of years. Careful excavation of this pertinent dataset divulges certain extremely interesting statistics. For example, analysis of dataset for a particular time interval, say most recent 6 months reveals that; there will be set of certain hot topics which will have more academic and research interest than other topics. Such novel topics can be generally termed as trending hot topics. The major drawback of this dataset is that it doesn’t provide mapping of article titles to various topics in computer science discipline and is not up to date. Hence it is very laborious and time consuming to manually find topics of article titles and performing topic based trends. Needless to say that, accuracy of clustering based hot topic detection will be very low. The novel framework proposed in this article employs computer science domain ontology to classify research paper titles in dynamically updated bibliographic dataset into respective topics which greatly enhances the accuracy and efficiency. ACM CCS Domain ontology which is well-known domain ontology has been used as an information model encompassing various hierarchical relationships between topics in computer science domain for this mapping.

Multiple advantages acquired by Ontology based detection of such hot topics in a temporal manner from dynamic bibliographic datasets can be. a) It provides consistent insights in identifying trending hot topics at a particular time period efficiently and quickly. b) It is possible to identify different dynamic emerging and bursty topic trends on a temporal basis.

- Domain ontology is always supported with a helper ontology with established and emerging terms for accurate detection of topic keywords. d) It is possible to recommend article titles to readers.

Map Reduce is a software framework to quickly process big unstructured data in parallel across a distributed
cluster of commodity computer systems. Because of its huge popularity, Map reduce libraries have been made available for different platforms. The open source implementation is part of Apache Hadoop framework. The proposed framework utilizes hadoop architecture and map-reduce algorithms to greatly enhance the speed while detecting hot topics, keywords and article suggestions.

The rest of the paper is organized as follows. Section 2 highlights the related works. Section 3 describes in detail on the novel framework used for the detection of hot topics based on domain ontology and map-reduce based cloud computing technique. Section 4 analyses the experimental results and discusses the performance measures and Section 5 concludes with future research directions.

2 RELATED WORK

Closed frequent keyword set [2] is generally used to detect topics rather than maximal frequent keyword-set. Most of the semantic information is lost when maximal frequent keyword set is used.

Clustering documents based on frequent itemsets [3] has been studied in the algorithms FTC and HFTC [4] and the Apriori-based algorithm [5]. Both of these works consider the documents as bags of words and then find frequent itemsets. The major lacuna of bag of words techniques or vector space model is that the relative ordering of terms are not maintained. Hence the semantic information present in the document is lost. In this proposed framework, phrases from the titles of the research papers are extracted first and frequent substrings are derived as frequent keyword-sets, maintaining the underlying semantics.

3 PROPOSED SYSTEM

The steps for proposed framework are listed below.

1. First extract titles of research paper from DBLP dataset using a DOM parser
2. Store it in a file.
3. Extract phrases and keywords from titles using map-reduce1 and without map-reduce.
4. A part of ACM classification ontology 2012 (topic-subtopic relation) is created using Protégé software and stored in owl format.
5. After keyword extraction, frequent keyword set is formed and stored in database on a temporal basis.
6. Also the frequent keywords are mapped with titles [1].
7. The frequent keyword set is then mapped with ACM classification ontology to find keyword’s relevant topic and subtopic, based on which trending topics are displayed.
8. sparql query with apache-jena api is used in java to map titles to topics.
9. A novel method is provided to retrieve titles under a topic or subtopics which are ranked based on a custom algorithm that identifies frequencies of frequent keywords from a title and uses that statistics for ranking.

The build system has following features:

1. A method to find trending subtopics under a broader topic.
2. Retrieval of research paper titles under a topic or a subtopic of a year (user must specify the year) and results are ranked based on custom ranking algorithm.
3. A user interface to list trending topics.

The proposed system has the following phases:

1. Pre-processing-This phase includes title extraction, phrase detection, keyword formation, frequent keyword identification using map-reduce and without map-reduce from xml dataset, mapping of frequent keywords to ontology and frequent keyword to title relation.
2. Trend detection and research paper title retrieval.

3.1 Title Extraction

DBLP dataset is a computer science Bibliography which contains the metadata of over 2.8 million publications written by many authors in several thousands of journals or conferences. DBLP dataset is an xml file which contains all bibliographic records. From DBLP dataset, DOM parser was used to extract titles and year. The extracted titles and year will be indexed and stored in file. Also, the DBLP dataset was processed using map-reduce framework for improving performance metrics.

3.1.1 Title Extraction – Map Reduce

Map-reduce framework processes large dataset in an efficient manner. Mapper function will take the input file and produces output as a set of titles based on year. Mapper function output is sorted. Map function will allocate the number of map tasks based on the number of input paths to the map function.

XML Configuration

1. Specify Input format class
2. Define start tag and end tag to be parsed
3. Specify Input path and output path
4. Specify all required class
Algorithm 1 - Title Extraction using Map Reduce

**Input to Map Function:** DBLP dataset

1) *Create a Java DOM XML parser*
2) *Get elements by tag name*
3) *For each node:
   i) retrieve Titles based on the year*

Algorithm 2 - Phrase Extraction using Map Reduce

**Input to Map Function:** Extracted Titles and Stop words

*Declare hash map<string, integer>*
*Declare string temp, phrase*
*Read stop words and store it in hash map*
*for each titles:*
   *Iterate until a space*
   *Store iterated word in temp*
   *check whether temp is in hash map*
   *if temp is in hash map*
   *eliminate temp*
   *store phrase*
   *else*
   *phrase += temp*
   *temp = “”*

Algorithm 3 - Keyword set Formation using Map Reduce

**Input to Map Function:** Extracted Phrases

*Declare string keyword*
*for each phrases:*
   *separate phrases by space*
   *for each word:*
   *keyword = word*
   *store keyword*

3.2 Phrase detection and Keyword formation

From each title, phrases are found out. A phrase can be defined as a substring present between two stop words. We used around 114 Standard English stop words. A keyword is a substring of phrase so that, underlying semantics is maintained. The keywords both 1-gram and bi-gram extracted will be associated with title in file. Similar to that of Title Extraction, Map-reduce is used to extract phrases and keywords from titles extracted.

![Architecture of the proposed Trend Detection System](image-url)
Example:

**Title:** DDBJ new system and service refactoring.

**Year:** 2012

**Phrases:** DDBJ new system, service refactoring

**Keywords (one gram):** DDBJ, new, system, service, refactoring

**Keywords (two gram):** DDBJ new, new system, system service, system refactoring.

Thus each keyword extracted will have potential to identify a topic or subtopic.

### 3.3 Frequent Keyword identification

**Algorithm 4: Frequent keyword set formation**

**Input:** Keywords (one gram or two gram)

1. Declare hashmap <string, int>
2. for each keyword:
   - if hashmap.get(keyword) = not found
     - hashmap.put(keyword,1)
   - else
     - frequency = hashmap.get(keyword)
     - hashmap.put(keyword,frequency + 1)

The closed frequent keywords have the potential to identify the trending topics.

The keywords from titles stored in file are used to form frequent keyword sets. Thus, the frequent keywords identified both one gram and two gram will be stored in database.

### 3.4 Keyword to ontology mapping

Using apache jena api with java, sparql query can be used to query ontology owl file. A sparql query is used to obtain the classification hierarchy and the result is stored in database. After storing of ontology, frequent keywords are mapped with topic and subtopics of ontology based on corresponding keywords from ontology.

### 3.5 Frequent keyword, titles mapping

Frequent keywords are then mapped with titles which contain the frequent keyword (Algorithm 5). In this study we have considered keywords with frequency of 20 or greater to be potential trends.

**Algorithm 5:** Frequent keyword (both one gram and two gram) to paper mapping

1. map = all frequent keywords from database
2. for each title in database:
   - keywords[] = get corresponding keyword of title
   - index = index no of title from database
   - for each keyword:
     - if keyword found in map
       - append(keyword, title)

This mapping will help in retrieval of paper titles based on a keyword, topic or subtopic. So the pre-processing helps to populate the backend, so that trends and paper retrieval can be easily done.

### 3.6 Trend detection and titles retrieval

Researchers are always on the lookout for finding trending topics and titles from their domain of interest. It is extremely cumbersome to manually sort and find the trends even if hot words are detected. Hence trend detection of titles is a must based on hot keywords. There are many existing methods for detecting trend. In this framework Trend detection is found by statistical frequency based methods. Based on the frequency, hot topics are found out. This part is designed to have a UI (User Interface) that list the trends and also list the papers based on input topic obtained from user. 1 gram 2 gram trends of keywords are provided from a domain and article titles are displayed based on these. Also, there is an option to display the trending subtopics belonging to a broader topic. The titles retrieved based on the hot keywords must be arranged using some ranking methods to effectively list the retrieved titles. Hence the titles retrieved are ranked based on the following algorithm (Algorithm 6) before being displayed.
Algorithm 6: Title ranking

**Input:** Topic, keyword or subtopic

- `papers[]` = obtain indexes of paper mapped with keyword
- `result<string,int>`

```
for each index in paper:
    title = getTitle(index) from database(db)
    keyword = getKeyWord(index) from db
    frequency = 0

for each keyword:
    frequency += get frequency of current keyword count from db

result.put(title, frequency)

sort result based on frequency in descending order
```

Thus the ranking algorithm uses the closed frequent keyword set to rank the paper.

**TABLE 1. General Trends**

<table>
<thead>
<tr>
<th>General Trends(One gram)-past 3 years</th>
<th>General Trends(Two gram)-past 3 years</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wireless (2271)*</td>
<td>Wireless sensor (739)*</td>
</tr>
<tr>
<td>Networks (2163)*</td>
<td>Cognitive radio (499)*</td>
</tr>
</tbody>
</table>

Fig. 2. Top trending topics based on analysis of 2013 papers

*-indicates the keyword frequency in dataset considered
4 PERFORMANCE

In the pre-processing stage we have used 6 node hadoop cluster and map reduce algorithms along with normal java based approach up to keyword formation for calculating and comparing performance metrics.

![Fig. 3. Search result for papers published under domain 'networks' for year 2013](image)

![Fig. 4. Year wise Comparative Performance during title extraction, phrase and keyword extraction](image)

**Hadoop Cluster Configurations**

- **Master**: Rami node, Secondary Name node, Job Tracker
- **System**: Intel core-i7 2.3 GHz processor, 8GB ram, 1TB harddisk
- **Slaves (6)**: Data Node, Task Tracker
- **System**: Intel core-i7 2.3 GHz processor, 8GB ram, 1TB harddisk
Relative ordering is not maintained in the map-reduce output, since the output gets sorted. So map-reduce can be used in situations where relative ordering is not required. Since relative ordering plays a major role in our approach, we opted for normal approach.

We used cox-stuart trend analysis method to find out the accuracy of trends. We obtained a value of 0.125, the value is greater than 0.05 (we chose a significance level of 95%). Therefore there is no significant trend. If value is less than 0.05 we accept the hypothesis as significant trend.

5 CONCLUSIONS

In this work we focused on a novel ontology based approach to quickly identify all the hot topics at a particular time interval by mapping article titles to topics from big xml bibliographic data set. We pre-processed the titles from DBLP dataset using hadoop map reduce and converted it into keywords which are then used to form frequent keyword sets. Subsequently it is used to find trending topics. ACM classification ontology helps to accurately match a title with a topic. Frequent keyword sets help in ranking of research paper titles. This framework aids budding researchers in quickly finding trending topics, subtopics and retrieve paper titles under their area of interest with high precision. As a future work trend detection based on hadoop based multi document summarisation and ontology is planned.

REFERENCES

[4]. F. Beil, M. Ester, and X. Xu, "Frequent Term-based Text Clustering," Proceeding of the 8th International Conference on Knowledge Discovery and Data Mining (KDD), 2002.

Swaraj K P is a research scholar in the Department of Computer Science and Engineering at Anna University, Chennai. He is the corresponding author of this paper. He can be contacted at swarajkp@cs.annauniv.edu

Dr. Manjula, D. is working as Head and Professor in the Department of Computer Science and Engineering at Anna University, Chennai.

Hemnath K B, Manish Kumar L, Adhithyan V are final year students pursuing B.E