Hadoop based Text Mining System for Identification of Chemicals Associated with Disease of Interest

Kritika Bhowmik
Pimpri Chinchwad College of Engineering,SSPU
Nigdi,
Pune-411044.

Tejal Aher
Pimpri Chinchwad College of Engineering,SSPU
Nigdi,
Pune-411044.

Vaibhav Kale
Pimpri Chinchwad College of Engineering,SSPU
Nigdi,
Pune-411044.

K. Rajeswari, PhD
HOD, Pimpri Chinchwad College of Engineering,SSPU
Nigdi,
Pune-411044.

M. Karthikeyan, PhD
Scientist, Digital Resource Centre, National Chemical Laboratory.
Pune-411008.

ABSTRACT
With huge amounts of biomedical data being generated day by day extracting statistical information about the chemicals mentioned in such huge databases manually is tedious and time consuming. Our system is mainly designed for naive users, which aims to automate data collection and knowledge extraction from chemical literature in a user friendly and efficient way on the hadoop platform. The system downloads the abstracts related to the disease of interest from Pubmed database. The text of the abstracts is then extensively parsed for chemicals such as protein/gene names and chemical compound names and classified into different classes. This analysis would prove to be helpful in various biomedical and pharmaceutical industries. The extraction of important information will be done using the Ling Pipe API wherein a training dataset is given to this Ling Pipe which classifies the extracted bioentities in the respective classes. The system being deployed on hadoop platform provides a scalable and distributed system which processes huge number of abstracts in a short time and with high efficiency. The system also provides a user friendly user interface for easy use of the hadoop system for non technical users.

General Terms
Map reduce based system, chemical extraction, chemical classification

Keywords
Text mining, chemicals, hadoop, LingPipe, bioentities, data extraction, data classification.

1. INTRODUCTION
Many times researchers need to manually search large amounts of text in the form of published articles in order to obtain pertinent information regarding proteins, genes and compounds related to a particular disease. This can be done manually as follows: i) Reading all the text documents, ii) Searching for the chemicals, iii) Classifying the chemicals and deriving the statistical data. Such a procedure can be very tedious and time consuming. Due to the exponential growth of data files produced by experiments, we need an automated system for extracting this information[2]. Since data arrives in abundance, using traditional centralized approach to extract knowledge from data is no longer used due to their limited ability to handle massive data. Thus a new scalable distributed framework has emerged: Hadoop[5], an opensource framework designed for storage and large scale processing work on a cluster of commodity hardware. Hadoop is based on MapReduce[6], a programming model for producing large scale parallel analysis. Thus we use the hadoop framework for the extraction of chemicals from the abstracts. Currently a top-to-bottom approach is used in most of the text mining applications, which means parsing important words from vast amounts of text, and then spread downwards through databases or software such as STRING[7], STITCH[8, 9], Biotextquest[10], Bioalma[11], Whatizit[12, 13] or iHop[14].Other tools, reviewed in[11], try to analyze literature using semantic concepts, clustering algorithms or ranking algorithms and most generic attempts go one step further by using semantic web concepts to further enrich and annotate terms found in literature[1]. In the proposed system the user will just need to provide the name of the disease of interest. The system then downloads abstracts related to the disease and stores it in the hadoop distributed file system. All these files are then submitted to hadoop job. This job extracts the chemicals from the document by parsing the text word by word and sending each word to the LingPipe tool. LingPipe tool then extracts the chemicals from the input words and classifies them in different classes. The number of occurrences of each chemical in a class for the respective disease is then represented in the form of graphs for each class and also a network of the chemicals is displayed.

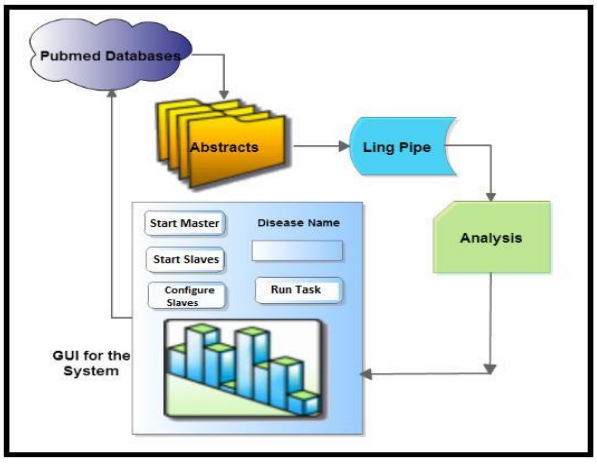

Fig. 1. Process of extracting chemicals from abstracts

2. SYSTEM PROCESS DEVELOPMENT

This work primarily aims at extraction of chemicals and their classification for the statistical analysis of different chemicals relevant to a disease. The figure 1 depicts the entire process.

Consider an example where the user gives the input disease of interest as Malaria. The system then downloads all the abstracts from the Pubmed database and all the chemicals occurring in these abstracts are extracted classified and their hit count is generated. The following is a sample text from Pubmed database with interferon-gamma classified as a protein.

The quantification of single cell interferon-gamma (IFN-γ) release for assessing cellular immune responses using the Enzyme-linked immunospot (ELISPOT) assay is an invaluable technique in immunology. Peripheral blood mononuclear cells (PBMC) are stimulated in vitro with recombinant proteins, peptides and recently whole malaria organisms.

The output of the system will be a graph of chemical names against their frequency as shown in figure 2 for the class of proteins:

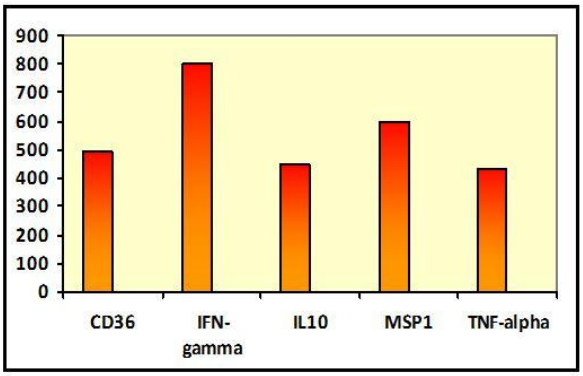

Fig. 2. Frequency count graph of proteins occurring in abstracts related to Malaria.

The development flow is as follows:

2.1 The Study of Hadoop Framework

Hadoop is a distributed framework used for processing BigData. Traditional centralised approach for data storage and processing is inefficient when data is in abundance. With the current rate of data being generated a component based scalable, fault taulerant, self recoverable, always available framework is needed which is provided by Hadoop. Considering distributed systems, traditional distributed systems are complex for programming. Data becomes a bottleneck because exchanging data needs synchronization. Hadoop is an opensource framework which uses the core concept of distributing the data as it is in the system. There is no need of transferring data in the initial stages of processing. The two core components of hadoop are:

2.1.1 Hadoop Distributed File System:

HDFS stores the data on the hadoop cluster (set of machines which have installed HDFS and Map Reduce). It sits on top of native file systems like ext3, ext4, etc. The data is first split into blocks and distributed across multiple nodes in the cluster. Each such block is first replicated by a replication factor as specified or by default 3 as shown in figure 3. HDFS stores files which are write once. If any update is to be done a new copy is stored in the HDFS. NameNode, DataNode and Secondary NameNode are the HDFS daemons. NameNode runs on the master node and stores the metadata. Secondary NameNode monitors the NameNode. DataNode actually stores the data blocks on the slave nodes. Each slave has a data node daemon running in the background.

2.1.2 Map Reduce

Map reduce is a method of distributing the tasks to all the nodes in a hadoop cluster. The core concept is to bring the code to the data rather than bringing the data to the code. This is done by Map Reduce. Each node thus processes data stored on that node. A map reduce job is given to a software daemon JobTracker which is responsible for the overall distribution and successful execution of the job. It assigns Map and Reduce tasks to the software daemons called TaskTracker running on each node of the cluster. Each TaskTracker is responsible for actually starting the map or reduce task and reporting the result back to the JobTracker. These daemons can be separated into two categories: The master node runs: NameNode, Secondary NameNode and JobTracker daemons. The slave runs: DataNode and TaskTracker daemons.

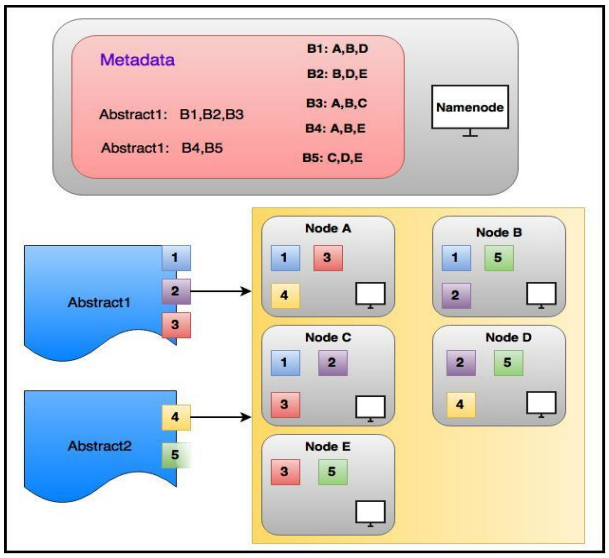

Fig. 3. Storing files in HDFS.

2.1.2.1 Mapper

The mapper reads data in the form of key value pairs. It outputs zero or more key value pairs.

Mapper (in key,in value) →(inter key,inter value)list. The mapper can use or completely ignore the input key. The standard method is to read one line at a time from a file. Where the key is the byte offset of that line in the file and the value is the line itself. The mapper then outputs the intermediate key value pairs.

2.1.2.2 Reducer

Once the map phase is over all the intermediate keys for a particular intermediate key from all the mappers are combined together in one list. Each such list is then given to a reducer. There can be single or multiple reducers based on the configuration. But all the values associated with a particular intermediate key are given to the same reducer. The intermediate keys and their values list is passed to the reducers in a sorted key order called as shuffle and sort an internal feature of hadoop map reduce framework. The reducer finally writes the output key value pairs to the HDFS.

2.2 The Study of Lingpipe Tool for Classification

LingPipe is a tool kit for processing text using computational linguistics. It provides a Java API for processing text. It provides features like:

- Tokenization
- Named Entity Recognition
- Identifies Significant Phrases
- Database Text mining
- Clustering

It is one of the most widely used open source tool for processing and classifying data. The royalty version of LinPipe is a free version provided by Alias-i which can be used for mining text and also classify it[16]. GENIA corpus, a package which comes with LingPipe, can be used for clustering of data. Internally it uses Named Entity Recognition algorithms for extracting the chemicals TokenshapeChunker is a LingPipe Named Entity Recognizer which can be trained by a training data set for retrieving the chemicals and classifying them into 36 different classes[16].

2.3 Proposed System Design

The system will consist of the following subsystems:

2.3.1 Collection subsystem

The user will mention the disease of interest and the collection sub system has to download all the relevant abstracts from the Pubmed database and load it on the Hadoop Distributed File System. As soon as the abstracts will be downloaded they will be divided into blocks, replicated and stored on the HDFS (Hadoop Distributed File system) as shown in the figure 3 After this, the MapReduce job starts processing these abstracts which is handled by the Classification Subsystem.

2.3.2 Classification susbsystem

The classification subsystem is responsible for mining the chemicals from the input abstracts classify them with the help of LingPipe segregating them into different files with respect to their classes and then calculating the frequency of occurrence of each chemical. All this is done by running two map reduce jobs on hadoop, described as follows.

Map Reduce phase 1

The master assigns the first map task to each node where input to the mapper is one line from the file at a time. Each line is then tokenized into words and passed through the LingPipe. The LingPipe then recognizes the chemicals from the input words and classifies them. In order to recognize the chemicals a dictionary is to be formed containing a list of chemicals. The LingPipe is trained with this dictionary for the classification. Finally after recognizing the chemicals the LingPipe gives the class name for the chemical. Thus the mapper emits the class name as the output intermediate key and the chemical name as the intermediate value. After the shuffle sort phase all the chemicals belonging to the same class are sent to the same reducer. Now each reducer writes the chemicals it receives to a file, named by the class name, ending the first map reduce phase.This is illustrated in the figure 4.

Map Reduce phase 2

Now, the master runs the second map task for each class name file. The mapper emits the name of the chemical as the key and 1 as the value for each instance of a chemical. After the shuffle and sort phase all the instances of a chemical that is 1 is aggregated as shown in the figure 5. Now each chemical with its intermediate values is sent to a reducer. The reducer sums all the 1s thus giving the count or the frequency of the chemical occurring in the abstracts related to the disease of interest. The reducer then writes the frequency of the chemical against the chemical name in a file ending the second map reduce phase.

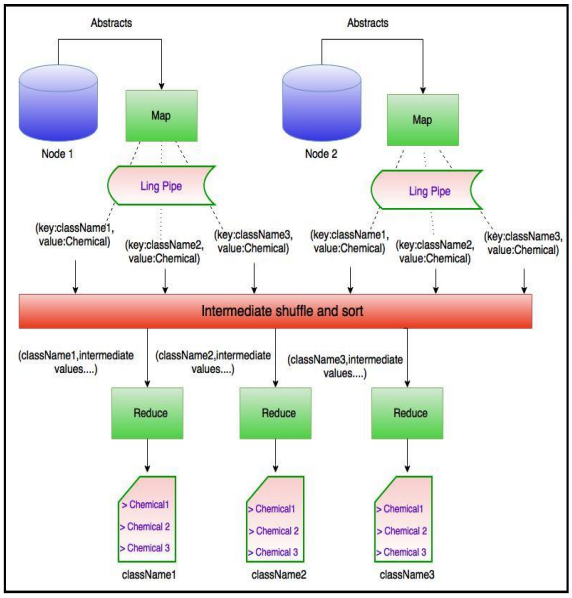

Fig. 4. Map Reduce Phase 1.

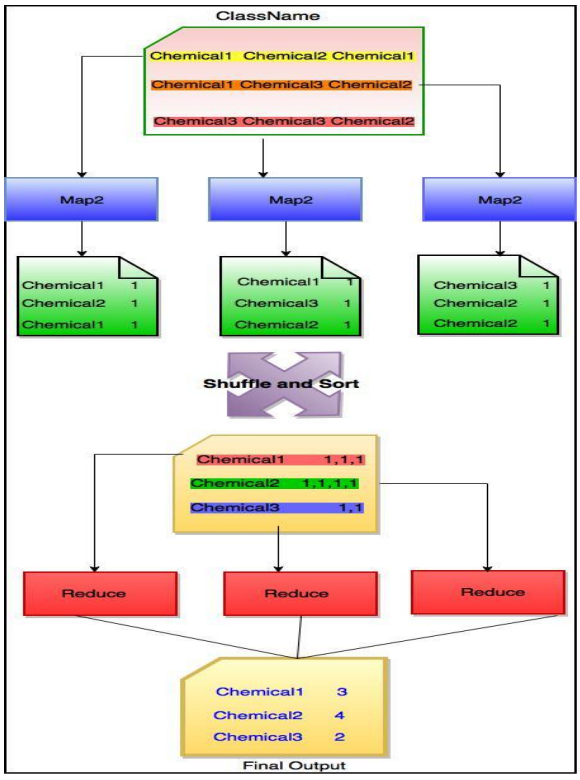

Fig. 5. Map Reduce Phase 2.

2.3.3 Analysis subsystem

The analysis subsystem takes the count of each chemical and

displays a graph for each class like protein, gene, etc, thus displaying 36 graphs and giving statistical information of the chemicals.

3. RELATED WORKS

Following are some of the works having similar functionality as the proposed system.

3.1 MegaMiner

It is a tool which segregates large amounts of data in parallel on cloud. Thus it is a distributed application which builds entity networks from given input files by parsing the data using LingPipe and MegaMiner libraries. It gives the hit count of the bioentities obtained and also the most frequently occurred terms[17].

3.2 OnTheFly 2.0

It is an automatic annotating and extraction tool. This web based application takes biological files as input and extracts bioentities from it and also provides knowledge about those entities stored in their database. It is a client server architecture based application which provides summaries, network of interaction of the bioentities and also pops up windows showing knowledge[16].

3.3 CheNER

It is a tool for automatically detecting and classifying chemicals by combing dictionary matching, regular expressions and linear CRFs in a different manner for classifying chemicals[18].

4. ADVANTAGES

Automated recognition of chemicals and their classification Distibuted system enabling the parallel processing of huge amounts of data with Hadoop platform. A graphical user interface for naïve users to easily configure the hadoop cluster, start the master and slave nodes and run tasks on hadoop. Graphical representation of statistical data for easy analysis of the chemicals extracted by the search.

5. FUTURE SCOPE

The project can be extended to further give information of the most frequently occurring chemicals and interrelation between them. We can also use this tool for analysing data of newly discovered diseases thus helping in the formulation of the medicines or vaccines of the same.

6. CONCLUSION

Extraction and classification of chemicals manually is a very cumbersome task. Thus there is need for automation of this work. The proposed system will hence be useful for researchers to parse extensively huge amounts of data with least efforts. The userfriendly graphical user interface enables non technical users to easily work on hadoop and process huge amounts of data in least time with great efficiency.

7. REFERENCES

[1] Evangelos Pafilis, Georgios A. Pavlopoulos, Venkata P. Satagopam, Nikolas Papanikolaou, Heiko Horn, Christos Arvanitidis, Lars Juhl Jensen, Reinhard Schneider, Ioannis Iliopoulos -OnTheFly 2.0: a tool for automatic annotation of files and biological information extraction.,IEEE,2013.

[2] Su Yan, W.Scott Spangler, and Ying Chen - Chemical Name Extraction Based on Automatic Training Data Generation and Rich Feature Set

[3] Spiros Papadimitriou ,Jimeng Sun DisCo: Distributed Coclustering with MapReduce A Case Study Towards Petabyte-Scale End-to-End Mining.

[4] Vincent Nicolas, Alzennyr Da Silva, Marie Luce Picard.- Heta: Hadoop Environment For Text Analysis.

[5] Tom White, Hadoop The Definitive Guide, OREILLY, 2009.

[6] Sanjay Ghemawat Jeffrey Dean, Mapreduce : Simplified data processing on large cluster, Google Inc, 2004.

[7] Von Mering, L. J. Jensen, B. Snel, S. D. Hooper, M. Krupp, M. Foglierini, N. Jouffre, M. A. Huynen, and P. Bork, STRING: known and predicted protein-protein associations, integrated and transferred across organisms, Nucleic Acids Res, vol. 33, no. Database issue, pp. D4337, Jan 1, 2005.

[8] M. Kuhn, D. Szklarczyk, A. Franceschini, M. Campillos, C. von Mering, L. J. Jensen, A. Beyer, and P. Bork, STITCH 2: an interaction network database for small molecules and proteins, Nucleic acids research, vol. 38, no. Database issue, pp. D5526, Jan, 2010.

[9] M. Kuhn, C. von Mering, M. Campillos, L. J. Jensen, and P. Bork, STITCH: interaction networks of chemicals and proteins, Nucleic acids research, vol. 36, no. Database issue, pp. D6848, Jan, 2008.

[10] N. Papanikolaou, E. Pafilis, S. Nikolaou, C. A. Ouzounis, I. Iliopoulos, and V. J. Promponas, BioTextQuest: a web-based biomedical text mining suite for concept discovery, Bioinformatics, vol. 27, no. 23, pp. 33278, Dec 1, 2011.

[11] R. A. Erhardt, R. Schneider, and C. Blaschke, Status of text-mining techniques applied to biomedical text, Drug discovery today, vol. 11, no. 78, pp. 31525, Apr, 2006.

[12] Rebholz Schuhmann, A. Jimeno Yepes, M. Arregui, and H. Kirsch- Measuring prediction capacity of individual verbs for the identification of protein interactions, Journal of biomedical informatics, vol. 43, no. 2, pp. 2007, Apr, 2010.

[13] D. Rebholz Schuhmann, M. Arregui, S. Gaudan, H. Kirsch, and A. Jimeno, Text processing through Web services: calling Whatizit, Bioinformatics, vol. 24, no. 2, pp. 2968, Jan 15, 2008.

[14] R. Hoffmann, and A. Valencia- A gene network for navigating the literature, Nature genetics, vol. 36, no. 7, pp. 664, Jul, 2004.

[15] Z. Lu- PubMed and beyond: a survey of web tools for searching biomedical literature, Database (Oxford), vol. 2011, pp. baq036, 2011. IEEE THIRD QUARTER 2007, Douglas O Shaughnessy.

[16] alias-i.com/lingpipe/ Ling pipe

[17] M. Karthikeyan and Renu Vyas, Practical Chemoinformatics.

[18] Anabel Usi, Joaquim Cruz, Jorge Comas, Francesc Solsona and Rui Alves CheNER: a tool for the identification of chemical entities and their classes in biomedical literature, 2015.