'FUTURIZM-2014, (14TH& 15TH March 2014), 'P.D.E.As College of Engineering, Manjari[Bk]', 'IIP Cell, 'Manjari Road, Hadapsar, Pune-412307. India.','Chief Editor:Dr.K.R.Harne', Editors: Prof R V Patil, Prof Niraja Jain'

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-2, Issue-1, October, 2014

Thin Mail Client

ShymBhujbal, PrashantBairagi, PrasadPingale, AvinashGaikwad

Ms. VidhyaNikam (Project Guide)

Department of Computer Engineering
University of Pune
{avinashgaikwad111,bshyam04,bairagiprashant1}@gmail.com

***Abstract*— In the Information retrieval, the main scuttle is, as the database goes larger in quantity or size, it will get increase in the search results of each search query. Then users have to search the information till he will not at the requested one or search one by one for every result by his own manually. This maximizes a lot of search done by the user and time for desired data. Eventually this approach for searching data is orthodox weary with search engines. But it will not in the favor of user's time and effort minimization strategy.**

So, for reducing the time consumption to get desired data or be at data set which useful for user's current area of work, there must be such mechanism which will search the data as per users given circumstances or search area, for this the concept of facets used in this system, to minimize all the above mentioned difficulties.

I. Introduction

The prime factors essential in our day to day life our time, data network and not to state but also a battery power. All these things go hand in hand. Say avoiding time, data usage in our mobile consumes a large volume of your mobile battery, and with less battery in certain situations doing certain things becomes time critical as mobile may go in offline mode. So thereby, all the three factors go coherently. The Email service provider allow to send data only upto 24 MB. Even with the help of google drive we can send data upto 10 GB, but still it's a constrain. The problem arises while sending a large volume of data in a short period of time through a mobile devices.

II. Proposed System

Now here considering a mobile device, sending a huge data in short time itself is a contradiction and here comes the sole and obsolete reason for this problem. The basic abstract of our program is Providing E-Mail Indexing Server and Thin Android Client for mailing large volume of attachments directly from user-defined server rather than uploading it and then sending through user-phone.

With this it will be possible to send a huge-sized data from our own mobile devices, but thinking in a other way around. There will be a pre-defined server where our all files will be stored already parsed and indexed. Basically we are supposed to send a query through our device to the server to send those files instead of first uploading it and then sending it.

The project study is focused on the concepts of Text Mining (Data Mining), Email Clustering, Lucene API, Web Services, JAVA and Computer Networks, operating over Android.

III. Web Services

Representational state transfer (REST) is an architectural style consisting of a coordinated set of architectural constraints applied to components, connectors, and data elements, within a distributed hypermedia system. REST ignores the details of

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-2, Issue-1, October, 2014

component implementation and protocol syntax in order to focus on the roles of components, the constraints upon their interaction with other components, and their interpretation of significant data elements.

REST has been applied to describe desired web architecture, to identify existing problems, to compare alternative solutions, and to ensure that protocol extensions would not violate the core constraints that make the Web successful. Fielding used REST to design HTTP and Uniform Resource Identifiers (URI).

The REST architectural style is also applied to the development of Web services as an alternative to other distributed-computing specifications such as SOAP.

The REST architectural style was developed by W3C Technical Architecture Group (TAG) in parallel with HTTP 1.1, based on the existing design of HTTP 1.0. The World Wide Webrepresents the largest implementation of a system conforming to the REST architectural style.

HTTP methods in Restful web services

HTTP Method	Operations Performed
GET	Get a resource
POST	Create a resource and other operations, as it has no defined semantics
PUT	Create or update a resource
DELETE	Delete a resource

Architectural Propertiesof Web Services

The Architectural properties induced by the architectural constraints of the REST architectural style are:

- Performance
- Scalability of component interactions

Fielding describes REST's effect on scalability thus:

REST's client–server separation of concerns simplifies component implementation, reduces the complexity ofconnectorsemantics, improves the effectiveness of performance tuning, and increases the scalability of pure server components. Layered system constraints allow intermediaries—proxies, gateways, and firewalls—to be introduced at various points in the communication without changing the interfaces between components, thus allowing them to assist in communication translation or improve performance via large-scale, shared caching. REST enables intermediate processing by constraining messages to be self-descriptive: interaction is stateless between requests, standard methods and media types are used to indicate semantics and exchange information, and responses explicitly indicatecacheability.

- Simplicity of interfaces
- Modifiability of components to meet changing needs (even while the application is running)
- Visibility of communication between components by service agents
- Portability of component deployment
- Reliability

IV. Apache Solr

Solr (pronounced "solar") is an open sourceenterprise search platform from the Apache Luceneproject. Its major features include full-text search, hit highlighting,faceted search, dynamic clustering, database integration, and rich document (e.g., Word, PDF) handling. Providing distributed search and index replication, Solr is highly scalable.Solr is the most popular enterprise search engine. Solr 4 adds NoSQL features.Solr is written in Java and runs as a standalone full-text search server within a servlet container such as Apache Tomcat orJetty. Solr uses the Lucene Java search library at its core for full-text indexing and search, and has REST-

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-2, Issue-1, October, 2014

like HTTP/XML andJSONAPIs that make it usable from most popular programming languages. Solr's powerful external configuration allows it to be tailored to many types of application without Java coding, and it has a plugin architecture to support more advanced customization.

V. Text mining

Text mining, also referred to as *textdemining*, roughly equivalent totext analytics, refers to the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning.

Text mining usually involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database), deriving patterns within thestructured data, and finally evaluation and interpretation of the output. 'High quality' in text mining usually refers to some combination of relevance, novelty, and interestingness. Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling (*i.e.*, learning relations between named entities).

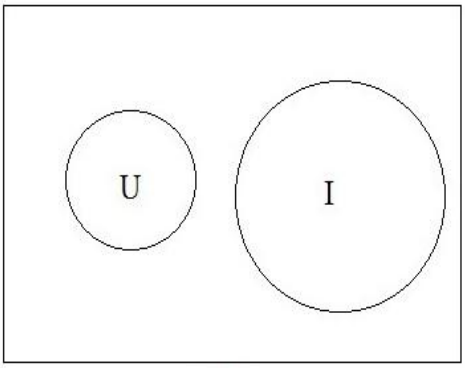
Text analysis involvesinformation retrieval, lexical analysisto study word frequency distributions,pattern recognition, tagging/annotation, information extraction, data mining techniques including link and association analysis, visualization, and predictive analytics. The overarching goal is, essentially, to turn text into data for analysis, via application of natural languageprocessing (NLP) and analytical methods.

VI. Algorithm

Consider for one user only

Let D(I) be the set of all documents Indexed in Server

And D(U) be the set of document to be indexed for user

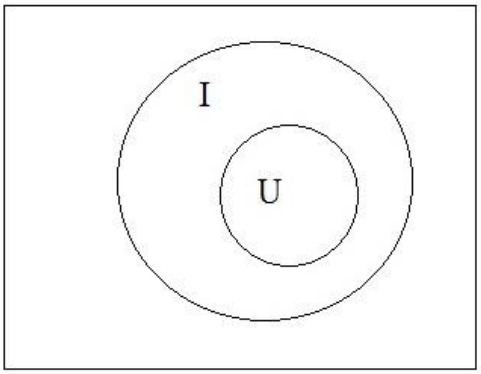


Initial

I:Index document on server

U:User's documents

After indexing



Indexing

Set of all document

D(I) U D(u)

Time required to return the documents

T(X) depends directly on D(X)

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-2, Issue-1, October, 2014

Where

X is the User

T(X) is Time required to search document

D(X) is document set of user X

For Faceting, The system has to distribute the documents into different groups.

So, Time required for Faceting also depends on size of set D(X)

Method for Searching a document.

Given Query submitted by user X is Q

Find the Documents D from Index which have terms present in Q

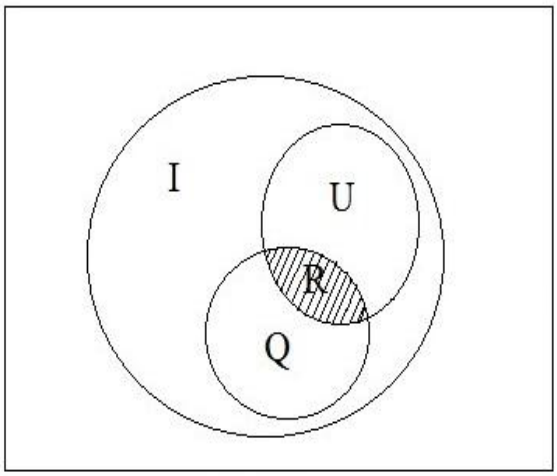
Iterate through all the documents D

Check if current Document Di is in G(X)

i.e. if current document belongs to user X

if yes , then add D(I) to ResultSet R.

Return the ResultSet R to User



Search
I: index document
U: user's document\
Q; query by user
R: result

Let D(I) be the set of all document

D(U) be the set of document belonging to user U

D(Q) be the set of indexed document having query Q

Result= R = D(U) ∩ D(Q)

Therefor

Time α D(U)

Time α D(Q)

Hence, Time α total document of D(I)

- Here, the time required to search the documents is proportional to set of indexed documents.
- Time required for faceting also proportional to set of indexed documents, and
- The time required to obtain the results also depends on size of query entered.
- Thereby, all equations are to be polynomial. Hence we can say our project is P complete.

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-2, Issue-1, October, 2014

V SystemImplementation

The proposed system is based on Android platform .In proposed system also use Web Services, Solr and Apache Tomcat applications.

The proposed system is mainly categorized into three modules which are : Indexing data, Searching data and E-mail Utility using Thin Android Application. Indexing process includes parsing of the documents, indexing data and storing that data on file system. Apache Tika is the parser used for parsing the data. It includes parser for all types of data (like PDF-Parser, DOC-Parser, etc). After parsing the data, the data is transferred to indexer for indexing. Apache Solr is used for Indexing the data.

After indexing the data, the data is stored on file system in predefined user space. Every user is provided with authorization for indexing their data on server. After this process, the data is available for search.

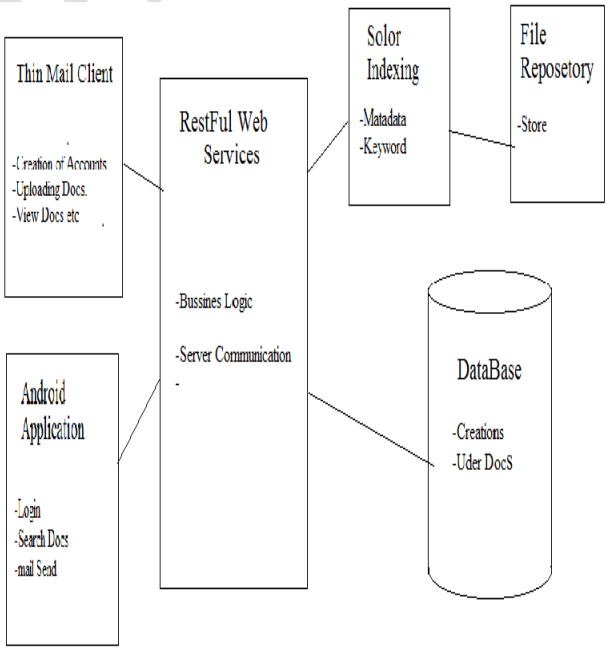
In the searching process, the system provides different kinds of Facets (like Field Facet, Range Facet and Content Facet) for filtering the data. User can search and go through the list of documents using some kind of keywords or name of the document to be searched. When user enter the name or keyword of document for search, the intended keyword with associated facet for filtering the data is passed to server. The server will process the query and filter data according to facets and returns the resultant documents to user.

The Android based E-mail utility is provided to share the data across Internet through E-mail. The biggest advantage is, User would be able to send email with attachments without actually the documents on the phone. Sometimes people want to send urgent emails and they do not have devices. Android driven emails can make their work lot easier, attachments can also be mailed without actually having them on Phone. It provides the Click-and-Send feature for sending the searched data.

VI Proposed Work:

Application Flow-

1. User creates his account on server.
2. User uploads all his documents (.pdf, .doc, .ppt, .xls, .mp3, etc) onto the server and stores it in the file system.
3. For faster retrieval and search , documents are parsed and kept indexed on Indexing Server.
4. User can then search for required documents by just typing the keywords/ search string. User can also perform faceted search, like getting the list of all docs of particular format (.mp3, .pdf, etc).
5. User can then mail that document using Android Application to the destination addresses.



Proposed System

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-2, Issue-1, October, 2014

Conclusion

Proposed application was able to identify documents successfully which contain common content to a set of documents inputted by the user. By using the application we were able to index different document formats such as doc, rtf and pdf while providing the capability for the user to add as many types of such formats as required in the future.

The initial objectives which were identified during the start of the project were successfully achieved. We were successfully able to extend the Lucene API and to show that it is possible to allow the user to select multiple documents and then retrieve documents which are similar to all those documents selected. Furthermore we were able show how two separate indexes can be compared against each other to find similarities between the two.

One can think of many more challenging research projects in the area of text mining and we only hope that we have convinced the reader that the study of this area is not only of great practical significance, but also promises to be an exciting area for further research.

References:

[1] I. Dagan. R. Feldman and H. Hirsh, "Keyword-based Browsing and Analysis of Large Documents Sets". In Proceedings of SDAIR-96, 5th Annual Symposium on Document Analysis and Retrieval. Las Vegas, UNLV Publications/Reprographics, Las Vegas: 191-207. 1996.

[2] M. Montes-y-Gomez, A. Gelbukh and A. Lopez-Lopez, "Mining the News: Trends, Associations and Deviations. Computation Systems 5(1): 14 – 25. 2001.

[3] R. Feldman, "Practical Text Mining". In proceedings of the 2nd European Symposium on Principles of Data Mining and Knowledge discovery. London: 478. 1998.

[4] R. Feldman and I. Dagan, Knowledge Discovery in Textual Databases(KDT). In proceedings of the 1st International Conference on Knowledge Discovery and Data Mining. Montreal, Canada. AAAI Press, Menlo Park, CA, 112-117.1995.

[5] http://poi.apache.org (16/10/2008)

[6] http://www.pdfbox.org/index.html (16/10/2008)

[7] Blake, C., and Pratt, W. (2001) Better Rules, Fewer Features: A Semantic Approach to Selecting Features from Text. In Proceedings of the 2001 IEEE International Conference on Data Mining. San Jose,CA, IEEE Computer Society Press, New York: 59–66.

[8] Hearst, M. (2003) What is Text Mining? http://www.ischool.berkeley.edu/~hearst/text-mining.html (29 October 2006).

[9] Gospodnetic, O., Hatcher, E. (2005) Lucene in Action. Manning.