RESEARCH ARTICLE OPEN ACCESS

Survey on Data Mining Algorithms for Daily Log System

Neha Jadhav[1], Vedant Bhavthankar[2], Manali Kamalaskar[3], Manleen Anand Kaur[4],
Computer Engineering,MIT College of Engineering,Pune-India
neha.j174@gmail.com[1],
bhavthankarvedant@gmail.com[2],
manalikamlaskar@gmail.com[3],
manleen.31dec92@gmail.com[4],

ABSTRACT

Log It is an android based application. It runs in the background, keeping track of everything that happens on phone. It is a type of a personal journal of the daily events and can also be thought of as analytics of life. This has access to a load of information, as such the permissions screen can seem a bit daunting, but it is needed in order to perform the functionality. Once the permission is given, it will take a few minutes to dig through the history and then provide with a dashboard insight of everything from emails to photos and even meetings that you've had. It is an end user system that keeps the log of person's daily usage of their cell phones which also helps to track the colleagues on route to their work stations. In our day to day life, we have no track of our daily activities. To know what we are up to every day, to learn from the mistakes that we did yesterday, and all such things, there should be a track to all our deeds. Not only to past, but one also should have reference to the present, what world is up to, where am I etc. All these questions need one master solution.

Keywords- K-means, Apriori

I. INTRODUCTION

Data mining is the process of discovering interesting patterns (or knowledge) from large amounts of data. The data sources can include databases, data warehouses, the Web, other information repositories, or data that are streamed into the system dynamically.

Data mining involves the use of sophisticated data analysis tools. These tools can include statistical models, mathematical algorithm and machine learning methods. Consequently, data mining consists of more than collection and managing data, it also includes analysis and prediction.

Data mining functionalities are used to specify the kind of patterns to be found in data mining tasks. In some cases, users may have no idea of which kinds of patterns in their data may be interesting, and hence may like to search for several different kinds of patterns in parallel. Thus it is important to have a data mining system that can mine multiple kinds of patterns to accommodate different user expectations or applications.

We will be using data mining technique to implement server side processing. The terminal device (here we will use Android phone) will send lot of information to the server. The relevant data will be sorted using data mining techniques.

Application will be like a personal diary where all our activities will be automatically captured and sent to a server. To get relevant information from this huge set of data we will use various data mining algorithms as per our need. K-means algorithm will be used to sort data with some parameters like date, name etc. Apriori algorithm will be used to find your colleagues, friends etc.

A message queue will be used at the server end to forward relevant messages to relevant people.

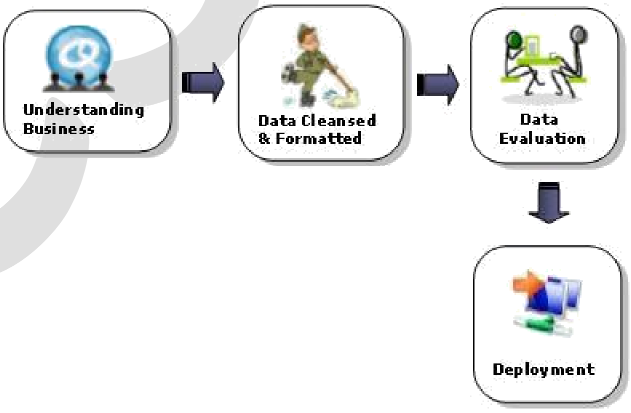

FIG 1: DATA MINING

II. DATA MINING TECHNIQUES

A. K-means Algorithm

K-means is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori.

The main idea is to define k centres, one for each cluster. These centres should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centre. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycentre of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centre. A loop has been generated. As a result of this loop we may notice that the k centres change their location step by step until no more changes are done or in other words centres do not move any more. Finally, this algorithm aims at minimizing an objective function know as squared error function given by:

$$J(V)=\sum_{i=1}^{c}\sum_{j=1}^{c_i}\left(\left\|x_i-v_j\right\|\right)^2$$

where,

'||xi - vj||' is the Euclidean distance between xi and vj.

'ci' is the number of data points in ith cluster.

'c' is the number of cluster centres.

Algorithmic steps for k-means clustering

Let X = {x1,x2,x3,……..,xn} be the set of data points and V = {v1,v2,…….,vc} be the set of centres.

1) Randomly select 'c' cluster centres.
2) Calculate the distance between each data point and cluster centres.
3) Assign the data point to the cluster centre whose distance from the cluster centre is minimum of all the cluster centres..
4) Recalculate the new cluster centre using:

$$v_i=(1/c_i)\sum_{j=1}^{c_i}x_i$$

where, 'ci' represents the number of data points in ith cluster.
5) Recalculate the distance between each data point and new obtained cluster centres.
6) If no data point was reassigned then stop, otherwise repeat from step 3).

Advantages

1) Fast, robust and easier to understand.
2) Relatively efficient: O(tknd), where n is # objects, k is # clusters, d is # dimension of each object, and t is # iterations. Normally, k, t, d << n.
3) Gives best result when data set are distinct or well separated from each other.

Working:
The system will collect the daily count of calls, messages, gallery files(music and photos) and GPS (Global Positioning System) locations in the database. Using k-means, each of this collected data will be clustered according to the timestamp. This bifurcated data will be displayed to the user with the help of GUI(Graphical User Interface) of the application.
This efficiently clustered data will then be used for solving the inefficacious problem of daily transportation

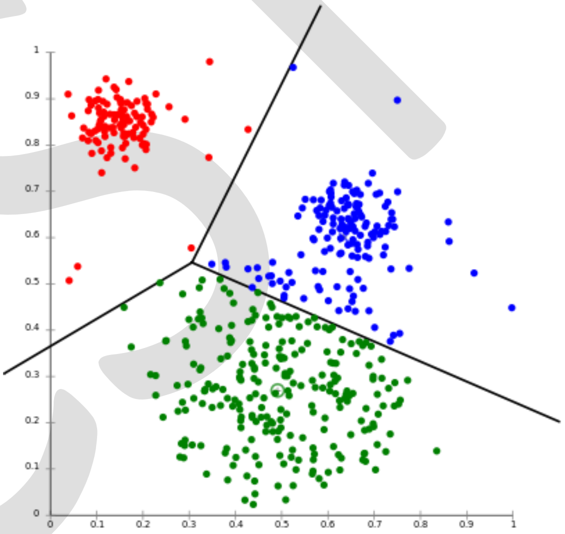

Fig 2 . Data Clusters

Apriori algorithm

```
Apriori($T,\epsilon$)
    $L_1 \leftarrow \{\text{large } 1 - \text{itemsets}\}$
    $k \leftarrow 2$
        while $L_{k-1} \neq$ emptyset
            $C_k \leftarrow \{a \cup \{b\} \mid a \in L_{k-1} \wedge b \in \bigcup L_{k-1} \wedge b \notin a\}$
            for transactions $t \in T$
                $C_t \leftarrow \{c \mid c \in C_k \wedge c \subseteq t\}$
                for candidates $c \in C_t$
                    $count[c] \leftarrow count[c] + 1$
            $L_k \leftarrow \{c \mid c \in C_k \wedge count[c] \geq \epsilon\}$
            $k \leftarrow k + 1$
        return $\bigcup_k L_k$
```

Explanation:

Apriori is a classic algorithm for frequent item set mining and association rule learning over transactional databases. It proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those item sets appear sufficiently often in the database.

Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as *candidate generation*), and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found.

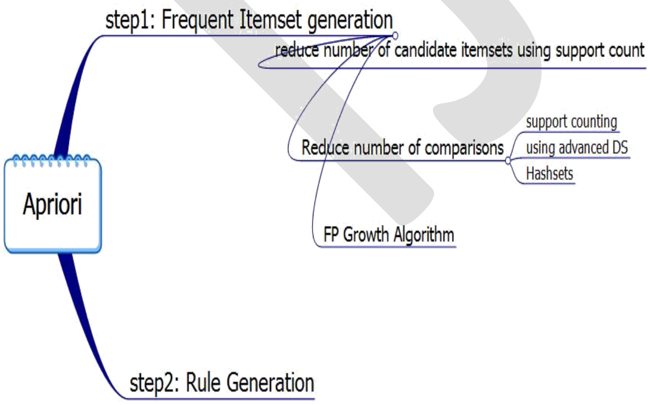
step1: Frequent Itemset generation
reduce number of candidate itemsets using support count
Reduce number of comparisons
support counting
using advanced DS
Hashsets
FP Growth Algorithm
Apriori
step2: Rule Generation

Fig 3. Apriori algorithm rule generation

Working:

Applying apriori algorithm on collected data, we can solve the problem as:

If A, B and C are three commuters travelling from paths a, b and c respectively, apriori algorithm will be applied to find out which combination of paths is the most efficient. The efficiency will be determined by the reduction in the resources used to commute by all three commuters collectively. Based on the combinations formed, the most efficient combination will be selected hence solving the inefficacious problem of daily transportation.

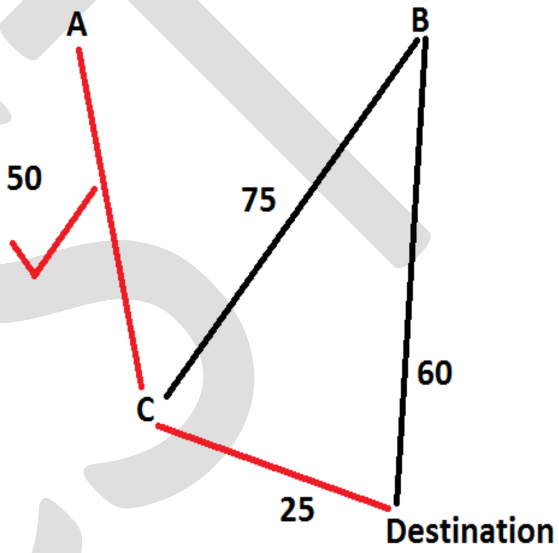

Fig 4. Example of Apriori Algorithm

III. Communication Protocols

B. RabbitMQ:

Messaging enables software applications to connect and scale. Applications can connect to each other, as components of a larger application, or to user devices and data. Messaging is asynchronous, decoupling applications by separating sending and receiving data.

RabbitMQ is a messaging broker - an intermediary for messaging. It gives your applications a common platform to send and receive messages, and your messages a safe place to live until received.

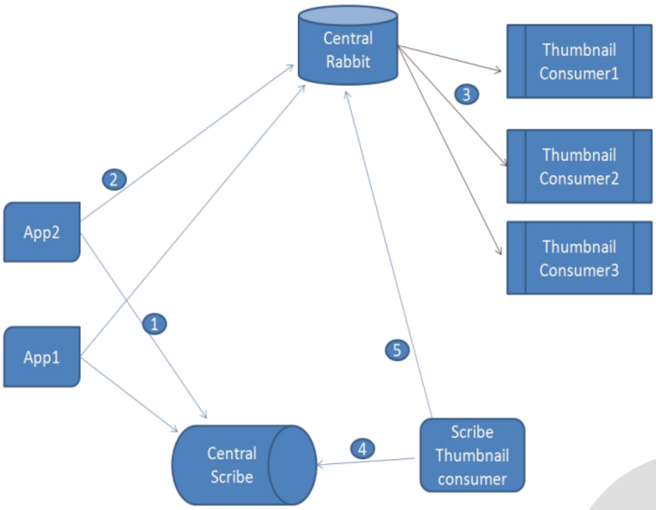

Fig.5. RabbitMQ architecture and working.

C. XMPP

Extensible Messaging and Presence Protocol (XMPP) is a communications protocol for message-oriented middleware based on XML (Extensible Markup Language). Unlike most instant messaging protocols, XMPP is defined in an open standard and uses an open systems approach of development and application, by which anyone may implement an XMPP service and interoperate with other organizations' implementations.

The original and "native" transport protocol for XMPP is Transmission Control Protocol (TCP), using open-ended XML streams over long-TCP connections.

An XMPP client is any software or application that enables you to connect to an XMPP for instant messaging with other people over the Internet. There are many free clients you can use to do this, for many different devices and operating systems.

The architecture of the XMPP network is similar to email; anyone can run their own XMPP server and there is no central master server. The Internet Engineering Task Force has formalized XMPP as an approved instant messaging and presence technology under the name of XMPP (the latest specifications are RFC 6120 and RFC 6121). No royalties are required to implement support of these specifications and their development is not tied to a single vendor. Custom functionality can be built on top of XMPP; to maintain interoperability, common extensions are managed by the XMPP Standards Foundation. XMPP applications beyond IM include group chat, network management, content syndication, collaboration tools, file sharing, gaming, remote systems control and monitoring, geo location, middleware and cloud computing, VoIP and Identity services.

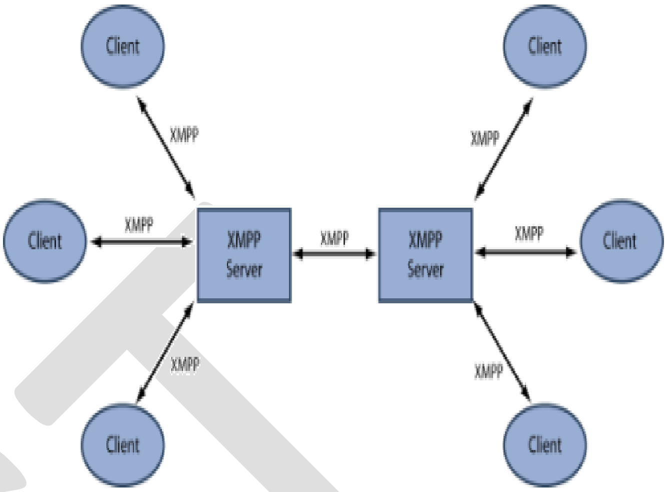

Fig 6. XMPP architecture

IV. CONCLUSION

The strategies mentioned as features of the application can become of great assistance to keep track of the most frequently used apps and files

-A list of phone contacts preferred (including SMS messages)

-How much time the user spends by the device per week, per month (here only the active time is to be considered, only when the user uses it)

- Travel path data through the city (using the phone GPS)

Thus this application provides the user with a detailed log so as to monitor all his activities at the click of a button.

REFERENCES

[1] Majdi Bsoul, Hlaing Minn, and Lakshman Tamil, *Apnea MedAssist: Real-time Sleep Apnea MonitorUsing Single-Lead ECG*, ieee transactions on informationtechnology in biomedicine, vol. 15, no. 3, may 2011

[2] Octavian Postolache,Pedro S. Girão*, Mário Ribeiro, Marco Guerra, João Pincho, *Enabling telecare assessment with pervasive sensing and Android OS smartphone*

[3] Pietro Albano, Aniello Castiglione, Giuseppe Cattaneo, Alfredo De Santis‡, *A Novel Anti-Forensics Technique for the Android OS*, 2011 International Conference on Broadband and Wireless Computing, Communication and Applications

[4] Demetrios Zeinalipour-Yazti, Christos Laoudias, Constandinos Costa, Michail Vlachos, Maria I. Andreou, Dimitrios Gunopulos, Crowdsourced Trace *Similarity*

with Smartphones, ieee transactions on knowledge and data engineering

[5] Kyosuke Nagata, Electrical Engineering and Electronics Saneyasu Yamaguchi, Hisato Ogawa, *A Power Saving Method with Consideration of Performance in Android Terminals*, 2012 9th International Conference on Ubiquitous Intelligence and Computing and 9th International Conference on Autonomic and Trusted Computing

[6] R.A.Ramlee, D.H.Z.Tang, M.M.Ismail, *Smart Home System for Disabled People Via Wireless Bluetooth*, 2012 International Conference on System Engineering and Technology September 11-12, 2012, Bandung, Indonesia

[7] Takayuki Matsudo, Eiichiro Kodama, Jiahong Wang, and Toyoo Takata, *A Proposal of Security Advisory System at the Time of the Installation of Applications on Android OS*, 2012 15th International Conference on Network-Based Information Systems

[8] George Meletiou1, 2, Ioannis Voyiatzis2, Vera Stavroulaki1, C. Sgouropoulou2, *Design and Implementation of an e-exam system based on the Android platform*, 2012 16th Panhellenic Conference on Informatics