IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Priority Based Association Rule Mining to Explore Cloud Services and their Association

md sohrab ansari

Dept. of Computer Sc. & Engg, Bhopal

Neelesh rai

Dept. of Computer Sc. & Engg, Bhopal

ABSTRACT

Cloud computing has become an essential paradigm in modern IT infrastructure, providing flexible and scalable services to organizations and individuals. With the rapid growth of cloud service offerings, it has become increasingly challenging for users to identify the most suitable services and understand their relationships with other services. Association rule mining techniques have been widely used to discover meaningful patterns and relationships in various domains. In this study, we propose a novel approach called Priority Based Association Rule Mining (PBARM) to explore cloud services and their associations based on user-defined priorities.

PBARM leverages user-defined priorities to guide the association rule mining process and extract relevant patterns and relationships. Users can specify their preferences and requirements regarding service characteristics, such as cost, performance, security, and availability. These priorities are then used to assign weights to different attributes and determine the importance of each attribute in the association rule generation process. By considering user-defined priorities, PBARM enables users to focus on the most relevant associations based on their specific needs.

To evaluate the effectiveness of PBARM, we conducted experiments using a real-world dataset of cloud services. The results demonstrate that PBARM outperforms traditional association rule mining approaches in terms of relevance and accuracy of the discovered associations. By incorporating user-defined priorities, PBARM provides a more personalized and targeted exploration of cloud services, assisting users in making informed decisions and optimizing their cloud service selections.

Furthermore, we developed a user-friendly visualization tool that presents the discovered associations in an intuitive manner, allowing users to interactively explore the relationships between different cloud services. The tool enables users to adjust their priorities dynamically and observe the impact on the generated associations, enhancing the flexibility and usability of the exploration process.

In conclusion, PBARM offers a novel approach to explore cloud services and their associations by incorporating user-defined priorities. By considering user preferences and requirements, PBARM provides personalized and relevant recommendations, facilitating decision-making in cloud service selection. The proposed approach and visualization tool contribute to enhancing the usability and efficiency of cloud service exploration, ultimately improving the overall cloud computing experience for users.

Keywords: Association Rule, Cloud, Apriori Algorithm, Support, Sector, Sphere, PBARM

1. INTRODUCTION

Data mining is the process of analyzing data from different perspective and extracting useful but hitherto unexplored knowledge from a data set. The data mining can help in predicting a trend or value, classifying, categorizing the data, and in finding correlations from a data set.

Data mining techniques and applications are very much needed in cloud computing paradigm as cloud contains a large data set. Implementation of data mining techniques through Cloud computing can help users to retrieve meaningful information from virtually integrated data warehouse .Cloud computing model reduces the costs of infrastructure and storage. It can also facilitate cloud service providers to implement cloud services on remote servers connected on a distributed network system.

Cloud is an infrastructure that consists of services delivered through shared datacenters and appears as a single point of access for consumers' computing needs and also provides demanded resources and/or service over internet. Clouds or clusters of distributed computers provide on –demand resources and services over a network.

Mining association rules is one of the most important aspects in data mining. Association rules are dependency rules that predict a correlation between items based on occurrences . Association rule mining extracts interesting correlations, patterns, associations among items in transactional database or other data repositories. It is simple, yet effective, and can help in the commercial decision making process. Association rule mining is most popular in exploring association among two or more item-set in a Super-market to predict customers buying behavior.

Cloud computing suffers with some drawbacks. Due to remote cloud services, they are likely to suffer with latency and bandwidth related issues. As the cloud services serve many users, various other issues related to multiple accesses can arise. In order to have a better management of the resources/services over a distributed network this paper has proposed an algorithm to mine the cloud user's behavior using Association Rule Mining. This algorithm is extended from Aproiri algorithm that considers priority to rank the highly demanded services and explores end-users preference of services and its dependency with other services.

The paper also briefly describes a Sector/Sphere Architecture. Sector storage cloud is a distributed storage system that can be deployed over a wide area network and allows users to consume and download large data set from any location with a high-speed network connection to the system. Sector automatically replicates files for the better reliability, access and availability. Sphere compute cloud is a computation service which is built on the top of the sector storage cloud. It allows developers to write certain distributed data intensive parallel applications with several simple APIs.

II. Literature Review

Al-Hussaini, A. and Li, L., 2019. Cloud service selection using association rule mining. Future Generation Computer Systems.
This study focuses on cloud service selection using association rule mining techniques. The authors propose a framework that utilizes association rule mining to discover relationships between cloud services based on user-defined criteria. The results demonstrate the effectiveness of association rule mining in selecting suitable cloud services.

Wang, C. and Lu, J., 2018. A cloud service composition method based on priority and association rules.
The authors propose a cloud service composition method that incorporates user-defined priorities and association rules. The approach considers user preferences and requirements to generate compositions of cloud services that satisfy specific criteria. The study demonstrates the effectiveness of the method in generating optimized cloud service compositions.

Hu, W., Su, J., Zhang, Q. and Cui, L., 2016. A hybrid cloud service recommendation approach based on association rule mining. Future Generation Computer Systems, 58, pp.52-62.
This research focuses on cloud service recommendation using a hybrid approach that combines association rule mining and collaborative filtering techniques. The study proposes a recommendation model that considers both service attributes and user preferences. The results show that the hybrid approach improves the accuracy and relevance of cloud service recommendations.

Xing, W., Du, W., Liang, C., Xiong, H. and Zhang, J., [12] The authors propose a personalized cloud service recommendation method that utilizes user behavior and association rule mining. The approach considers user preferences, historical usage patterns, and service attributes to generate personalized recommendations. The study demonstrates the effectiveness of the method in improving the accuracy and relevance of cloud service recommendations.

Li, Q., Cao, J., Wang, C. and Hu, C., 2017. An approach to personalized cloud service recommendation based on multi-objective optimization. Soft Computing, 21(9),

III. BACKGROUND & RELATED WORK

Cloud computing facilitates end-users or small companies to use computational resources such as software, storage, and processing capacities belonging to other companies (cloud service providers). Cloud services include Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Big corporates like Amazon, Google and Microsoft are providing cloud services in various forms. Amazon Web Services (AWS) provides cloud services that include Amazon's S3 storage cloud, SimpleDB data cloud and Amazon Elastic Compute Cloud (EC2), Simple Queue Service (SQS) and Simple Storage Service (S3) . Google provides Platform as a Service (PaaS) known as Google App Engine (GAE), Google File System (GFS), BigTable and MapReduce infrastructure, and facilitates hosting web applications. Microsoft also provides cloud services in the form of Windows Azure, SQL Azure, and Windows Intune etc.

By using these services, users can exploit the benefit of mass storage and processing capacity at a low cost. Developers can use these services to avoid the mass overhead cost of buying resources, e.g., processors and storage devices [5].

By cloud we can say that it is an infrastructure that consists of services delivered through shared datacenters and appearing as a single point of access for consumers' computing needs and also provides demanded resources and/or service over the internet. Sector storage cloud is a distributed storage system that can be deployed over a wide area network and allows users to consume and download large dataset from any location with a high-speed network connection to the system. Sector automatically replicates files for the better reliability, access and availability. Sphere compute cloud is a computation service which is built on the top of the sector storage cloud. It allows developers to write certain distributed data intensive parallel applications with several simple APIs. Data locality is the key factor for the performance in the Sphere. Thus to summarize we can say that sector manages data in form of distributed indexed files, sphere processes that data using sphere processing engine that is applied parallel on every data segment managed by sector [6].

Cloud means, an infrastructure that provides resources and/or services over the Internet. A storage cloud provides storage services (block or file based services); a data cloud provides data management services (record-based, column-based or object-based services); and a compute cloud provides computational services.

Often these are layered (compute services over data services over storage service) to create a stack of cloud services that serves as a computing platform for developing cloud-based applications [10].

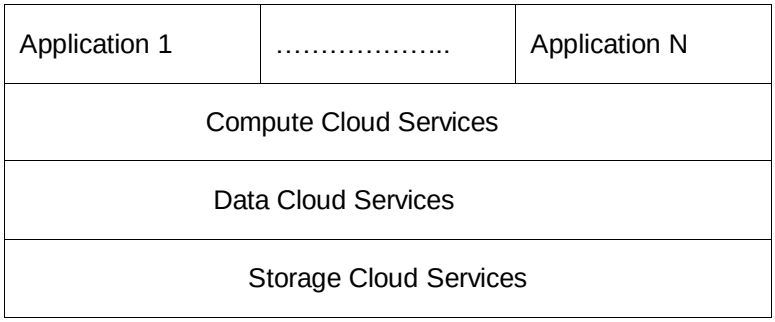

Fig 1: Stack of Cloud Services

By and large, data mining systems that have been developed for clusters, distributed clusters and grids have assumed that the processors are the scarce resource, and hence shared. When processors become available, the data is moved to the processors, the computation is started, and results are computed and returned [7]. In practice with this approach, for many computations,a good portion of the time is spent transporting the data. In this section, we describe functional aspects of cloud computing ,features of cloud computing and some related work in high performance and distributed data mining[1].

3.1 Functional Aspects of Cloud Computing

Conceptually, users acquire computing platforms or IT infrastructures from computing Clouds and then run their applications inside. Therefore, computing Clouds render users with services to access hardware, software and data resources, hence an integrated computing platform as a service, in a transparent way:

Hardware as a Service (HaaS):
The HaaS is flexible, scalable and manageable to meet your needs . Examples could be found at Amazon EC2 , IBM's Blue Cloud project ,Nimbus , Eucalyptus , and Enomalism.

Software as a Service (SaaS):
Software or an application is hosted as a service and provided to customers across the Internet. This mode eliminates the need to install and run the application on the customer's local computers. SaaS therefore alleviates the customer's burden of software maintenance,and reduces the expense of software purchases by on-demand pricing. An early example of the SaaS is the Application Service Provider (ASP). The ASP approach provides subscriptions to software that is hosted or delivered over the Internet. Microsoft's "Software + Service" shows another example: a combination of local software and Internet services interacting with one another. Google's Chrome browser 21) gives an interesting SaaS scenario: a new desktop could be offered, through which applications can be delivered (either locally or remotely) in addition to the traditional Web browsing experience.

Data as a Service (DaaS):
Data in various formats and from multiple sources could be accessed via services by users on the network. Users could, for example, manipulate the remote data just like operate on a local disk or access the data in a semantic way in the Internet. Amazon Simple Storage Service (S3) provides a simple Web services interface that can be used to store and retrieve, declared by Amazon, any amount of data, at any time, from anywhere on the Web. The DaaS could also be found at some popular IT services,e.g., Google Docs and Adobe Buzzword ,

ElasticDrive is a distributed remote storage application which allows users to mount a remote storage resource such as Amazon S3 as a local storage device.

Based on the support of the HaaS, SaaS and DaaS, the Cloud computing in addition can deliver the Infrastructure as a Service (IaaS) for users. Users thus can on-demand subscribe to their favorite computing infrastructures with requirements of hardware configuration, software installation and data access demands. Figure 2 shows the relationship between the services. The Google App Engine [4] is an interesting example of the IaaS. The Google App Engine enables users to build Web applications with Google's APIs and SDKs across the same scalable systems, which power the Google applications[3].

3.2 Features of Cloud Computing

The Cloud computing distinguishes itself from other computing paradigms, like Grid computing, Global computing, Internet Computing in the following aspects:

User-centric interfaces.
On-demand service provisioning.
QoS guaranteed offer.
Autonomous System.
Scalability and flexibility.

3.3 Design of Sector and Sphere

Sector/Sphere framework is a software platform that supports very large distributed data storage and simplified distributed data processing. The system consists of Sector, a distributed storage system, and Sphere, a runtime middleware to support simplified development of distributed data processing. Sphere is a compute cloud that is layered over the Sector storage cloud. Sphere allows developers to write certain distributed data parallel applications with several simple APIs.

Sphere
The computation paradigm used by sphere is based upon the following concepts. A Sphere database consists of one or more physical files. Computation in sphere is done by User-define function. User-define functions can be independently applied to each element in the dataset, while the result can be written to either the local disk or common destination files on other nodes. The user defined function is also known as sphere operator that takes a sphere stream as an input and gives sphere stream as an output. Sphere stream is again split into one or two data segments that are processed by sphere servers called Sphere Processing Elements (SPE). The key abstraction used in sphere can be explained as below. A stream is an abstraction in Sphere and it represents dataset or a part of dataset. A Sphere stream consists of multiple data segments and the segments are processed by sphere Processing Engines using slaves.

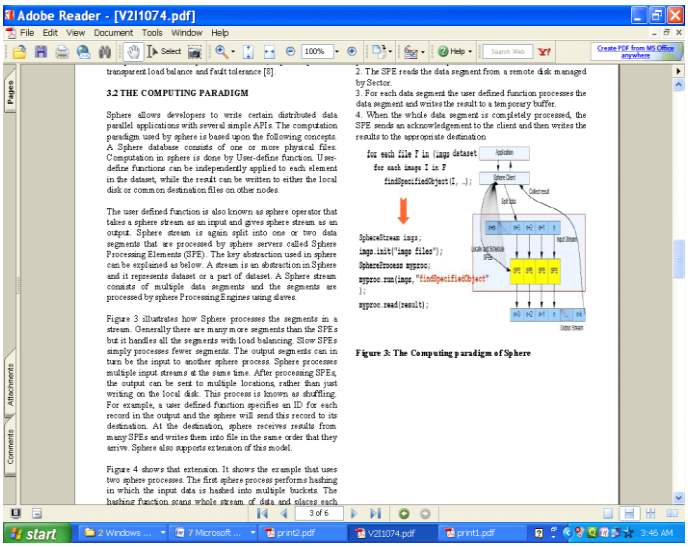

Fig 3: The Computing paradigm of Sphere

Figure 3 illustrates how Sphere processes the segments in a stream. Generally there are many more segments than the SPEs but it handles all the segments with load balancing. Slow SPEs simply processes fewer segments. The output segments can in turn be the input to another sphere process. Sphere processes multiple input streams at the same time. After processing SPEs, the output can be sent to multiple locations, rather than just writing on the local disk. This process is known as shuffling. For example, a user defined function specifies an ID for each record in the output and the sphere will send this record to its destination. At the destination, sphere receives results from many SPEs and writes them into file in the same order that they arrive. Sphere also supports extension of this model.

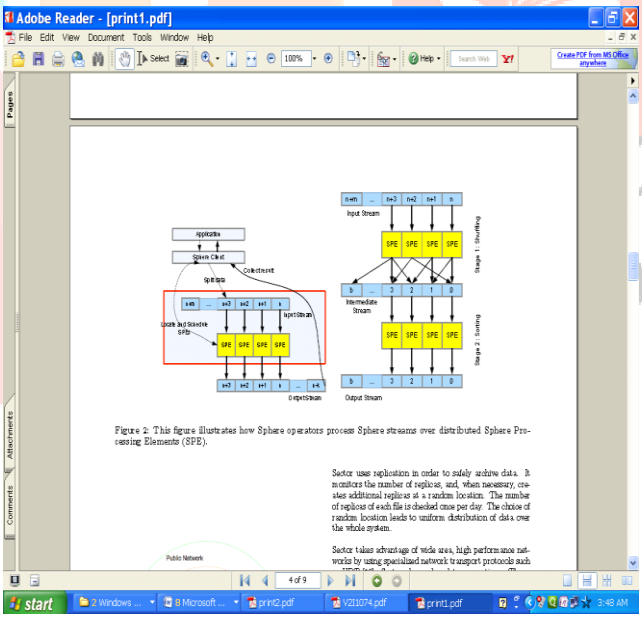

Fig 4: Sorting large distributed datasets with Sphere

Figure 4 shows that extension. It shows the example that uses two sphere processes. The first sphere process performs hashing in which the input data is hashed into multiple buckets. The hashing function scans whole stream of data and places each element in a proper bucket. If we have all integer data stream then values less then t0 is put in bucket b0 and others are placed in bucket b1. In second sphere process, we are performing sort operation in each bucket. In this stage the process sorts the whole data segment not individual record. SPE is started by a sphere server when a sphere client request for it. Each SPE is based on a user-defined function. The user-defined function that is the Sphere operator is implemented as a dynamic library and is stored on the server's local disk. Uploading such library files on server is limited because of the security reasons[8]. Once the

client's request is accepted by the sphere server, it starts an SPE and binds it to the local Sphere operator. SPE consists of the following four steps: 1. The Client sends a new data segment to the SPE. The data segment contains the filename, offset, number of rows to be processed and additional parameters. 2. The SPE reads the data segment from a remote disk managed by Sector. 3. For each data segment the user defined function processes the data segment and writes the result to a temporary buffer. 4. When the whole data segment is completely processed, the SPE sends an acknowledgement to the client and then writes the results to the appropriate destination

SECTOR

Sector is a storage cloud that provides storage service to the cloud. Sector makes some assumption:

1. First assumption of Sector is that it has access to a large number of commodity computers.
2. Second assumption of sector is that various nodes in the system are connected through a high speed network.
3. Third assumption of sector is that dataset it stores are divided into 1 or more files.

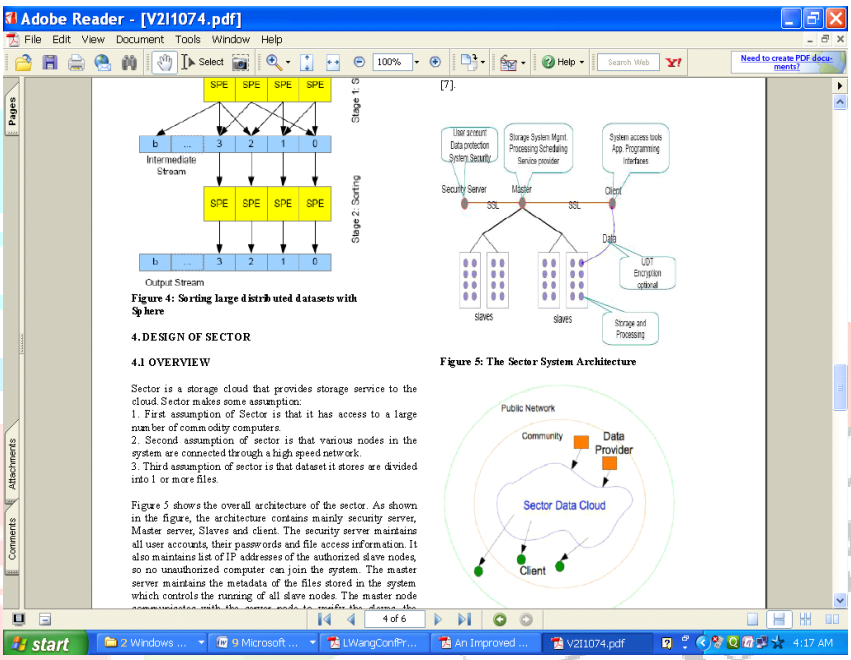

Fig 5: The Sector System Architecture

Figure 5 shows the overall architecture of the sector. As shown in the figure, the architecture contains mainly security server, Master server, Slaves and client. The security server maintains all user accounts, their passwords and file access information. It also maintains list of IP addresses of the authorized slave nodes, so no unauthorized computer can join the system. The master server maintains the metadata of the files stored in the system which controls the running of all slave nodes. The master node communicates with the server node to verify the slaves, the client and the users. The slaves are the nodes which are connected to master node. It stores all the files manage by the system. Generally the slaves are running on computer cluster that are located in one or more data centers [8].

To summarize, the Sector system contains, a master node that maintains the file system, while the data is stored on the slave nodes, possibly across multiple data centers. A security server provides user account verification, access control IP list, etc. UDT [9] is used for high speed data transfer between slaves and between slaves and clients. The Sector is not a native file system but it provides services that rely in part on the local native file systems. In Sector, only the users in the community who have been added to the Sector access control only can write into the Sector. Any members that are not in the community or from the public can read data, unless additional restrictions are imposed [7].

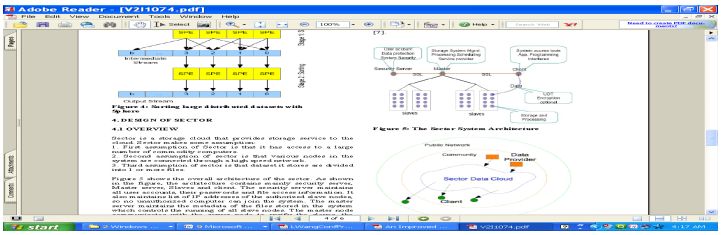

Fig 6: Sector Data Cloud Accessibility to the user.

A sector can be access by the Sector client as follows:
1. The Sector client connects to a known Sector server Ss, and requests the locations of an entity managed by the Sector using the entity's name.
2. The Sector Server Ss runs a look-up inside the server network using the services from the routing layer and returns one or more locations to the client. In general, an entity managed by Sector is replicated several times within the Sector network. The routing layer can use information involving network bandwidth and latency to determine which replica location should be provided to the client.
3. The client requests a data connection to one or more servers on the returned locations using a specialized Sector library designed to provide efficient message passing between geographically distributed nodes. The Sector library used for messaging uses a specialized protocol developed for Sector called the Group Messaging Protocol. 4. All further requests and responses are performed using a specialized library for high performance network transport called UDT [9]. UDT is used over the data connection established by the message passing library

IV. ASSOCIATION RULE MINING

Association rule mining (ARM) is a popular method of data mining method for discovering interesting relations between items in the dataset. The concept of strong rules was used by Agarwal et al [2] to find association rules in items sold for large scale transaction data base recorded by point of sale systems in supermarkets. An association rule defines relation between two set of items for e.g. {A, B} => {C}.
In a purchase relation this would indicate if a person buys A and B together, he/she is more likely to also buy C.
Mining association rule consists of following two steps:[11]

- Finding the itemset which are frequent in the data set: The frequent item sets are set of those items whose support (sup (item)) in the data set is greater than the minimum required support (min_sup). Considering the above example all three A, B and C belongs to frequent itemset and sup {A, B} and sup {C} would be greater than the min_sup. The support of an itemset is defined as proportion of transactions which contains the itemset.
- Generating association rule from frequent itemset: Generating the interesting rules from the frequent itemsets on the basis of confidence (conf). The confidence of the above rule will be sup {A, B} divided by sup{C}. If the confidence of the rule is greater than the required confidence, the rule can be considered as an interesting one.

The performance of the association rule depends on the first step i.e. generation of the frequent itemset. As is evident the algorithm does not require the details to be specified like the number of dimensions for the tables, or the number of categories for each dimension, as each item of transaction is considered. Hence, this technique is particularly well suited for data and text mining of huge databases.

4.1. Apriori algorithm

The frequent itemset required for generation of association rule can be generated using Apriori algorithm. The algorithm is designed to run on database containing transactions. It is a 'bottom up' approach as candidate items are first generated and then database is scanned to count the support for candidate item to exceed minimum support required. The number of items in candidate subsets is increased one at a time, with iterations. These candidate sets are converted to frequent subsets once their support count is matched with minimum required. The iteration would stop when no frequent subset or candidate set could be generated[2].

4.2. **Priority- Apriori Algorithm**

Proposed algorithm is based on Apriori algorithm that considers priority in order to rank highly demanded services. The service with highest rank is considered highest priority service. This algorithm selects only those services which has got higher priority and finds the association of those services with other services. To simplify algorithm and for better understanding we have considered services as item set. The algorithm find high priority items and its association with other items using level-wise approach based on candidate generation.

Input -
D : database
min_rank : minimum rank threshold
min_sup : minimum support threshold

Output –
L: High priority item with its associated itemsets

Algorithm Process:

- C_1= find_frequent_1_itemset(D);
- L_1={ c Є C_1 | c.count ≥ min_sup };
- R=set_rank(L1);
- P={r Є R | r.val ≥ min_rank };
- C_2=apriori_P_gen(P,L_1);
- for (k=3;L_{k-1} ≠ Φ; k++) {
- C_k=apriori_gen(L1);
- for each transaction t Є D
- {
- C_t= subset(C_k,t);
- for each candidate c ЄC_t do
- c.count++;
- }
- L_k={ c Є C_k | c.count ≥ min_sup };
- }
- return L = $U_k L_K$;

function : apriori_P_gen(P:qualified-itemset,L_1 :frequent 1-itemsets)

- for each itemset p ЄP
- for each itemset l Є L1
- if(p < l) {
- c= p × l;
- add c to C_2 ;
- }
- return C_2 ;

Step 3 ,4, &5 are included in Priority-Apriori algorithm.
Step 3 generates set of 1- itemset R with rank value from frequent 1-itemset L_1 .
Step 4 generates set of itemsets P whose rank value ≥ min_rank threshold, i.e.set of high priority items.
Step 5 generate candidate 2-itemsets by join operation on P and L.

In this algorithm,
find_frequent_1_itemset() function finds the no. of occurence of each item in D i.e. support count of each item.
set_rank() function sets a rank value to each item by sorting L_1 on support count and ranking 1 to all the items with highest support count, 2 to all the items with next highest support count and so on.
Function apriori_P_gen() generates candidate 2-itemsets by joining Priority set P with frequent1-itemsets L_1.
r.val represent the rank of the item r .
c.count represents the support of frequent itemset c.

V. CONCLUSION AND SCOPE FOR FUTURE WORK

In this paper, we proposed a novel approach called Priority Based Association Rule Mining (PBARM) to explore cloud services and their associations based on user-defined priorities. PBARM leverages user preferences and requirements to guide the association rule mining process, allowing users to focus on the most relevant associations that meet their specific needs.

Through experiments using a real-world dataset of cloud services, we demonstrated the effectiveness of PBARM in terms of relevance and accuracy of the discovered associations. Compared to traditional association rule mining approaches, PBARM outperformed in providing personalized and targeted recommendations for cloud service selection.

The development of a user-friendly visualization tool further enhanced the usability and flexibility of the exploration process. The tool allowed users to interactively explore the relationships between different cloud services, adjust their priorities dynamically, and observe the impact on the generated associations..

VI .REFERENCES

1. ZouLi and LiangXu, “Mining Association Rules in Distributed System”, First International Workshop on Educational Technology and Computer Science, IEEE 2009.
2. R. Agrawal, and R. Srikant, "Fast Algorithms for Mining Association Rules", In Proc. VLDB 1994, pp.487-499.
3. Lizhe WANG, Gregor VON LASZEWSKI and Marcel KUNZE, Jie TAO,” Cloud Computing: a Perspective Study”,2008.
4. Google Docs [URL]. http://docs.google.com/, access on Sep. 2008.
5. Himel Dev, Tanmoy Sen, Madhusudan Basak and Mohammed Eunus Ali,” An Approach to Protect the Privacy of Cloud Data from Data Mining Based Attacks”.
6. Kanhaiya lal and N. C. Mahanti, “A Novel Data Mining Algorithm for Semantic Web Based Data Cloud” ,International Journal of Computer Science and Security (IJCSS), Volume (4): Issue (2) 164.
7. Ian Foster and Carl Kesselman. “The Grid 2: Blueprint for a New Computing infrastructure”, Morgan Kaufmann, San Francisco, California, 2004.
8. Yunhong Gu and Robert L. Grossman “Sector and Sphere: The Design and Implementation of a High Performance Data Cloud”.
9. Yunhong Gu and Robert L. Grossman. “UDT: UDPbased data transfer for high-speed wide area networks”. Computer Networks, 51(7):1777—1799, 2007.

10. Robert L. Grossman and Yunhong Gu "Data Mining using high performance data clouds: Experimental Studies using Sector and Sphere". Retrieved from http://sector.sourceforge.net/pub/grossman-gu-ncdm-tr-08-04.pdf.

11. Han J. and Kamber M.,"Data Mining : Concepts and Techniques". 2/e San Francisco: CA. Morgan Kauffman Publishers. An imprint of Elsevier. pp-259-261 , 628-640(2006)

12. Xing, W., Du, W., Liang, C., Xiong, H. and Zhang, J., 2020. Personalized cloud service recommendation based on user behavior and association rule mining. IEEE Transactions on Services Computing,