International Journal of Scientific Research & Engineering Trends
Volume 1, Issue 4, July-2015, ISSN (Online): 2395-566X

Implementing Behavior Prediction using Frequent Pattern Tree using Data Mining Technique in Smart Home

Ms. Sonam Sheikh
Department of Computer Science & Engineering
Nagpur Institute of Technology, Nagpur,, India
Email: sonamsheikh0@gmail.com

Prof. Vaidehi Baporikar
Department of Computer Science & Engineering
Nagpur Institute of Technology, Nagpur, India
Email: vaidehiycce2008@gmail.com

***Abstract* – A fully fledged smart home system contains a wide variety of household appliances. With the popularity of smart home, production of large amount of data in our everyday life occurs. These existing data is used to analyze users' behavior and how to provide personalized services to different, seems to become a more and more important issue. We proposed a system that will gather all the information a human performed ,will apply good clustering that cluster all relevant information of the user. Here we are using FP growth Algorithm for predicting user's behavior on the basis of frequent data sets we have gathered in the form of their actions from every activity of registered user in Smart Home. On these activity we have applied An association Rule and can be able to predict particular's pattern behavior on the basis of calculating their average sentiment analysis.**

***Keywords* – Data Mining Approach, FP Growth Approach, User Behavior Analysis , Behavior Association Rules.**

I. INTRODUCTION

Data mining is the process of extracting interesting (non-trivial, implicit, previously unknown and potentially useful) information or patterns from large information repositories such as: relational database, data warehouses, XML repository, etc. Also data mining is known as one of the core processes of Knowledge Discovery in Database .

With the popularity of smart home, there will emerge sufficient data in our everyday life. Using traditional data analysis techniques to deal with the isomerization data set are having many limitations and how to use these data to analysis of user's behavior. We are making a system that will gather all the information in the form of actions performed by registered user or frequent user in Smart Home .Then will apply clustering that cluster all relevant information of the user. After that we perform association rules that generates the association on the gathered data . This gathered data will be responsible to predict the behavior of that user. Prediction will going to perform using calculation of Average Sentiment of multiple items and gives efficient result providing score and characteristics behavior of that person.

II. BEHAVIOR PREDICTION TECHNIQUES

The popularity of Smart Home, is emerging sufficient data in day to day life. Many people begin to pay attention to make full use of the data processing capabilities of smart devices to analyze data of smart appliances to extracts the user's normal patterns of life and the habits, and finally provide users with personalized service and remind how to use these existing data to analyze user's behavior and how to provide personalized services to different people under the specific conditions such as a particular time or place, seems to become a more and more important issue.

III. FUNCTIONAL DESCRIPTION

We are proposing a system that will gather all the information a human performed. In this system we will apply good clustering that cluster all relevant information of the user. After that we will perform efficient association rules that associate a gathered data.

In this system, smart home is a home which we are using for predicting the nature of human by activity, he/she performed on electronic device with the help of information gathering in order to predict the his/her behavior according to that information .Firstly we form cluster of data using k-means algorithm. After association we will apply some fuzzy theory that will predict a human behavior regarding human task that he/ she performed on difference devices according which will predict behavior of user using fuzzy logic. And finally we will analyze the result.

1. K -means approaches and algorithm

For achieving proposed work we apply a k-means algorithm for clustering data. We will use genetic algorithm for forming association rules over the cluster. And we will apply some fuzzy theory to predict the human behaviour. K-means algorithm is used for clustering data. It means clustering is a method of vector quantization, originally from signal processing ,that is popular for cluster analysis in data mining. K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.

International Journal of Scientific Research & Engineering Trends
Volume 1, Issue 4, July-2015, ISSN (Online): 2395-566X

2. Generating Association Rules

There are two important basic measures for association rules, support(s) and confidence(c) .Since the database is large and users are concerning about only those frequently purchased items, usually thresholds of support and confidence are predefined by users to drop those rules that are not so interesting or useful. The two thresholds are called minimal support and confidence respectively. Additional constraints of interesting rules also can be specified by users the two basic parameters of Association Rule Mining support and confidence.

Support(XY) = Support count of XY / Total number of transaction in D

Confidence of an association rule is defined as the percentage/fraction of the number of transactions that contain XUY to the total number of records that contain X, where if the percentage exceeds the threshold of confidence an interesting association rule X=Y can be generated.

Confidence(X|Y)= Support(XY)/Support(X)

IV. Association Rule Mining Approaches

Association Rule mining is a well explored research area. There are again some basic and classic approaches for association rule mining as stated ,the second sub problem of association rule Mining is straightforward .Most of those approaches Association rule mining is a well explored research area, There are some basic and classic approaches for association rule mining ,as stated , the second sub problem of ARM is straightforward, most of those approaches focus on the 1rst sub problem. The 1st sub problem can be further divided into two sub problems: candidate large item sets generation process and frequent item sets generation process.

4.1 Apriori Series approach:

The AIS(Agrawal, Imielinski, Swami) algorithm was the first algorithm proposed for mining association rule. It focus on improving the quality of databases together with the necessary functionality to process decision support queries. In this algorithm, only one item consequent association rules are generated which means that the consequent of those rules only contain one item, for example we only generate rules like,

X∩Y→Z

But not those rules as;

X→Y∩Z

To make this algorithm more efficient, an estimation method was introduced to prune those itemsets candidates that have no hope to be large, consequently the unnecessary effort of counting those itemsets can be avoided. Since all the candidate itemsets and frequent itemsets are assumed to be stored in the main memory, memory management is also proposed for AIS when memory is not enough. One approach is to delete candidate itemsets that have never been extended. Another approach is to delete candidate itemsets that have maximal number of items and their siblings, and store this the parent itemsets in the disk as a seed for the next pass. The main drawback of the AIS algorithm is too many candidate itemsets that finally turned out to be small are generated, which requires more space and wastes much effort that turned out to be useless. At the same time this algorithm requires too many passes over the whole database.

4.2 Apriori Algorithm:

Apriori is a great improvement in the history of association rule mining. The AIS is just a straightforward approach that requires many passes over the database, generating many candidate itemsets and storing counters of each candidate while most of them are turn out to be not frequent. Apriori is more efficient during the candidate generation process for two reasons, Apriori employs a different candidates generation method and a new pruning technique. There are two processes to find out all the large itemsets from the database in Apriori algorithm. First the candidate itemsets are generated, then the database is scanned to check the actual support count of the corresponding itemsets.

In the process of finding frequent itemsets, Apriori algorithm avoids the effort of wastage in counting the candidate itemsets which are known to be infrequent. These candidates are generated by joining among the frequent itemsets level-wisely and also candidates are pruned according to the Apriori property. As a result, the number of remaining candidate itemsets which are ready for further support checking becomes much smaller, which dramatically reduces the computation, I/O cost and memory requirement.

Algorithm:

Input:
database D
Mini Support ϵ
Mini Confidence ζ
Output:
Rt All association rules
Method:
01 *L*1 = large 1-itemsets;
02 for(k=2; *Lk¡*-1 ≠ ϕ 6= ; k++) do begin
03 *Ck* =apriori-gen(*Lk¡*1); //generate new candidates from *Lk¡*1
04 for all transactions T ϵ D do begin
05 *Ct*=subset(*Ck*,T); //candidates contained in T.
06 for all candidates C € *Ct* do
07 Count(C)=Count(C)+1; // increase support count of C by 1
08 end
09 *Lk*={C ϵ *Ct* | Count(C) ≥ϵ × | D| }
10 end
11 *Lf*=U*k* *Lk*;
12 *Rt*=GenerateRules(*Lf* , ζ)

To break the two bottlenecks of Apriori series algorithms, some works of association rule mining using tree structure have been designed.

4.3 Frequent Pattern Tree Approach:

FP-Tree, frequent pattern mining, is another milestone in the development of the association mining, which aims to break the bottleneck of the Apriori. The frequent items sets are generated with only two passes over the databases and without any candidate generation process. FP-Tree was introduced by Han. By avoiding the candidate generation process and less passes over the database, FP-Tree is an order of magnitude faster than the Apriori algorithm. The frequent patterns generation process includes two sub processes: constructing the FP-Tree, and generating frequent patterns from the FP-Tree

The process of constructing the FP-Tree is as follows;

1. The database is scanned for the 1rst time, during this scanning the support count of each items are collected. As a result the frequent *1* -itemsets are generated. This process is the same as in Apriori algorithm. Those frequent item sets are sorted in a descending order of their supports.

2. Create the root node of the FP-Tree T with a label of Root. The database is scanned again to construct the FP-Tree with the head table, for each transaction the order of frequent items is resorted according to the head table.

The frequent patterns are generated from the FP-Tree by the procedure named FP-growth. Based on the head table and the FP-Tree, frequent patterns can be generated easily. The FP Growth method indexes the database for fast support computation via the use of an augmented prefix tree called the frequent pattern tree (FP-tree). Each node in the tree is labeled with a single item, and each child node represents a different item.

4.4 Algorithm : FP-Growth:

Input: A database DB, represented by FP-tree constructed according to Algorithm 1, and a minimum support threshold .

Output: The complete set of frequent patterns.

Method: call FP-growth(FP-tree, null).

Procedure FP-growth(Tree, a)

{

(01) if Tree contains a single prefix path then // Mining single prefix-path FP-tree

(02) {

(02) let P be the single prefix-path part of Tree;

(03) let Q be the multipath part with the top branching node replaced by a null root;

(04) for each combination (denoted as ß) of the nodes in the path P do

(05) generate pattern ß ∪ a with support = minimum support of nodes in ß;

(06) let freq pattern set(P) be the set of patterns so generated;

}

(07) else let Q be Tree;

(08) for each item ai in Q do { // Mining multipath FP-tree

(09) generate pattern ß = ai ∪ a with support = ai .support;

(10) construct ß's conditional pattern-base and then ß's conditional FP-tree Tree ß;

(11) if Tree ß ≠ Ø then

(12) call FP-growth(Tree ß , ß);

(13) let freq pattern set(Q) be the set of patterns so generated;

}

(14) return(freq pattern set(P) ∪ freq pattern set(Q) ∪ (freq pattern set(P) × freq pattern set(Q)))

}

The efficiency of FP-Tree algorithm account for three reasons first is compressed representation ,secondly only scans the database twice , thirdly FP-Tree uses a divide and conquer method.

V. Calculating the Average Sentiment of Multiple Items

Here the behavior of an individual is calculating using sentiments calculating formula. Because sentiment scores for individual documents are highly dependent on the amount and nature of the content in each document, it is not efficient to apply pure math operations across sentiment scores. So Averaging multiple sentiment scores can give a skewed overall picture. Instead of looking at aggregate counts and what they represent is what is recommended. We've developed a "P/N ratio" formula that gives average sentiment score across documents. The PN ratio calculation formula is aimed for getting an average sentiment for the group of the records with sentiment scores.

The formula in pseudo code is:

```
IF positive_count < negative_count
THEN
        ratio = -1*(negative_count/positive_count)

IF positive_count > negative_count
THEN
  ratio=positive_count/negative_count
                 ELSE
ratio=0
      IFratio<-10THENratio=-10
      IF ratio > 10 THEN ratio = 10
```

VI .Result

By applying proposed scheme and algorithm i.e FP tree approach ,we are getting a better cluster that contain related information. And better association over that cluster and a technique that predict the good result of human behavior.

In this system, we are providing efficient result than existing system. We have used FP Growth Algorithm for association which is generating better result than Apriori Algorithm. We are predicting behavior of user or Frequent User by analysis of frequent data set. And we are providing three cases for behavior predicted. In each case, depending upon his / her information collected by central database we have shown the classification of behavioral pattern in below three cases. Using the Average Sentiment analysis formula and Threshold Value considered in FP growth Algorithm this classification have taken place.

- Case I Normal
- Case II Happy
- Case III Sad

REFERENCES

[1] Mav Home Smart Home Project. Available online: http://ailab.wsu.edu/mavhome/index.html

[2] CASAS Smart Home Project. Available online: http://www.ailab.wsu.edu/casas/ (accessed on 16 February 2013).

[3] Frequent Pattern Growth (FP-Growth) Algorithm An Introduction-By Florian Verhein, January 10, 2008

[4] Multicore-enabling FP-tree Algorithm for Frequent Pattern Mining: *Posted by Yuxiong , Jun 25, 2009*

[5] Association Analysis: Basic Concepts andAlgorithms: By Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison-Wesley.

[6] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. *In Proc. of SIGMOD*, pages 207–216, 1993.

[7] Han Jiawei, Kamber Miceline. Fan Ming, Meng Xiaofeng translation. Data mining concepts and technologies, Machinery Industry Press,2001.

[8] Agrawal R.Imielinski T.Swami A. Mining association rules between sets of items in large databases, *Proceedings of the ACM SIGMOD Conference* on Management of data, pp. 207-216, May,1993.

[9] J. Han, J. Pei, and Y. Yin, " Mining Frequent Patterns without Candidate Generation", *Proc. 2000 ACM-SIGMOD Int. Conf. on Management of Data (SIGMOD'00),2000.*

[10] O. Zaiane, M. El-Hajj, and P. Lu. Fast Parallel Association Rules Mining without Candidacy Generation. *In IEEE 2001 International Conference* on Data Mining (ICDM'2001), 2001.

[11] Liu, L., Li, E., Zhang, Y., and Tang, Z. 2007. Optimization of frequent itemset mining on multiple-core processor. *In Proceedings of the 33rd international Conference* on Very Large Data Bases.2007

[12] H. D. K. Moonesinghe, Moon-Jung Chung, Pang-Ning Tan. Fast Parallel Mining of Frequent Itemsets. Michigan State University.

[13] C. Borgelt. An Implementation of the FP-growth Algorithm. Workshop Open Source Data Mining Software. 2005

[14] Frequent Pattern Growth (FP-Growth)Algorithm An Introduction by Florian Verhein.

[15] Multicore-enabling FP-tree Algorithm for Frequent Pattern Mining by Yuxiong

AUTHOR'S PROFILE

Ms Sonam Sheikh

Pursuing M.Tech. (CSE) from Nagpur Institute Of Technology, Nagpur, RTMNU University. Received bachelor's degree in 2012 from JDCOE, Nagpur. Having experience of 2.5 years as Assistant Professor Author's in NIT, Nagpur, RTMNU university.

Ms Vaidehi Baporikar

Received M.E GHRCE from, RTMNU University. She had done her graduation from YCCE, Nagpur at RTMNU. She is having experience of 5 years as Assistant Professor and Currently working as Asst. Prof. at NIT, Nagpur in RTMNU university.