Integration of SQL Modelling and Graph Representations to Disaggregated Human Activity Data for Effective Knowledge Extraction

Ravi Kanth Motupalli [1], Dr. O. Naga Raju[2]

1 Reasearch Scholar, Dept. Of CSE, Acharya Nagarjuna University, AP - email: ravikanth51@gmail.com

2 Assistant Professor& Head, Dept. Of Computer Science, Govt. Degree College, Macherla, AP

ABSTRACT

With high migration of technology to smart living attributes such as smart homes, smart devices, automatic appliances, and activity trackers, and etc, the data processing and analysing are much needed to get the high outcomes with less power usage and clear understanding of usage information. For this multiple author were proposed different ways and even we proposed an effective disaggregated data modelling for behavioural analytics using distributed architectures. Where implementing distributed architecture is first time in the industry, our model has lot of advantages including, high availability, security to data stored and can achieve high speed of huge number data storage and processing with easy integration of open source technologies. In this paper, we are implementing a model for activity labelling based on human actions and enhancing our earlier research model with interfacing the SQL models like Apache Hive and Google Big Query for easy processing with simples queries and making processing and analysing to reach more users with less efforts along with development of effective pictorial representations of resultant data with graph models built by PHP libraries.

Keywords:

Data Modelling, Smart Homes, Energy Disaggregation Data, Human Activity, SQL Modelling, Knowledge Extraction

Article Received: 18 October 2020, Revised: 3 November 2020, Accepted: 24 December 2020

I. INTRODUCTION:

Here majorly we are performing disaggregation towards the energy usage of power consumption data for generalising the usage statistics and clear understanding of power utilisation. Here we are initially collecting energy consumption data from multiple sources and multiple channels and combining with aggregation parameters for easy and public understanding of utility consumption. But we are dividing this aggregated data into individual datasets based on usage attributes for clear understanding and deep analysis of individual appliance or device usage. In our earlier research of energy consumption and statistic modelling we been used the UK-DALE dataset where it is first open source, high accuracy data consumption details published in United Kingdom in the year of 2012. As it is already disaggregated with power usage of individual appliances, we are taking this as input and performing various analytical models with respect to time, day, appliance type, human actions, and human behaviour aspects.

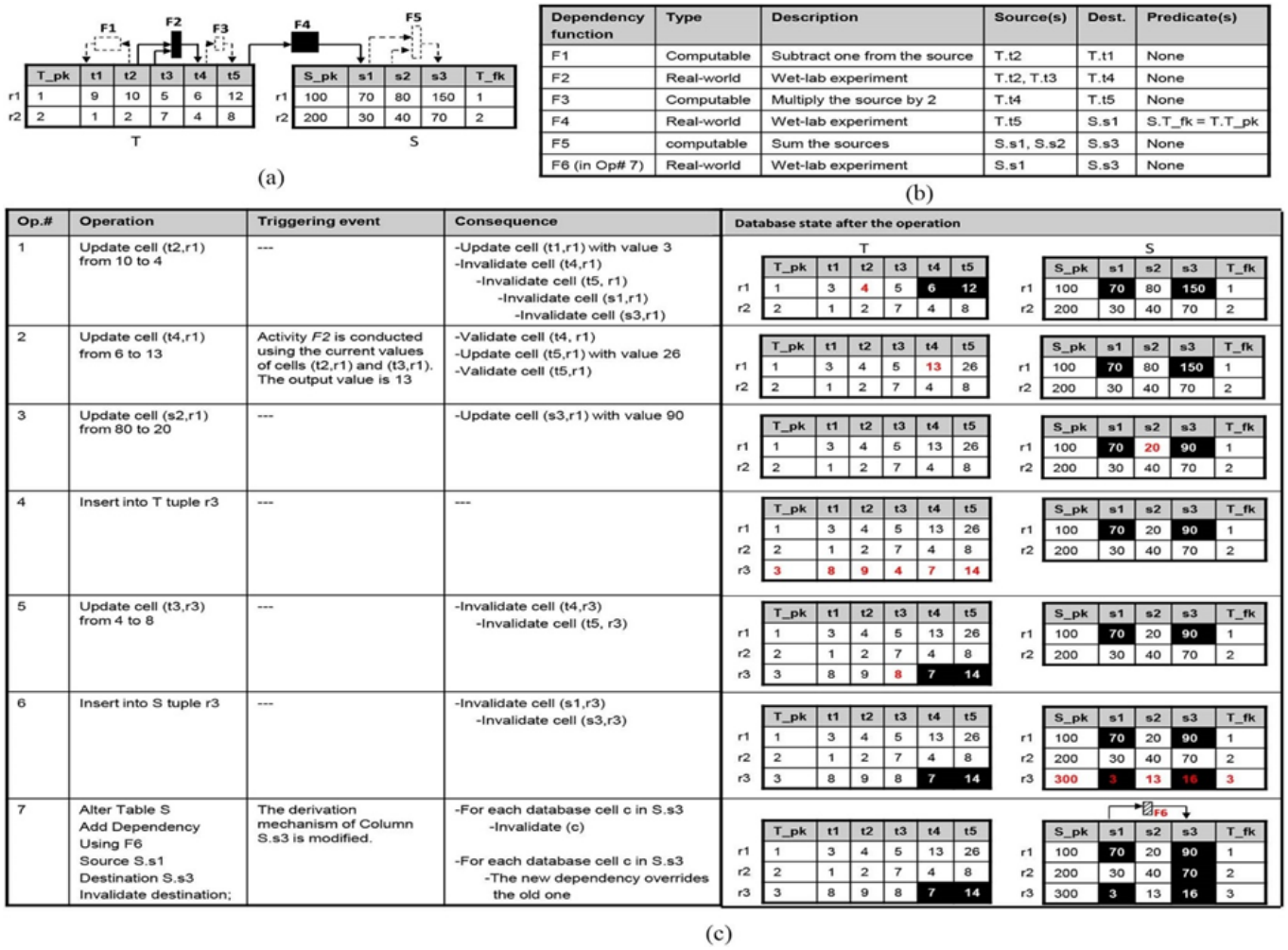
In the mission to plan and actualize a superior vitality disaggregation framework, specialists require a few kinds of information: The essential prerequisite is for datasets which record the force request of entire homes as well as the 'ground truth' power request of individual apparatuses inside the home. In 2011, specialists at MIT discharged the primary open dataset for vitality disaggregation research [2]. From that point forward, eleven more datasets have been discharged. These datasets have been generally welcomed at the same time, in light of the fact that each dataset utilizes an alternate document group, it is tedious to import different datasets. This is an issue in light of the fact that a significant rule for assessing any AI calculation is the way well it sums up over different datasets. A further test with existing datasets is that machine-lucid metadata is frequently insignificant and utilizes a pattern and jargon one of a kind to that dataset. Best case scenario, the absence of a standard metadata pattern makes it tedious to compose programming to process various datasets.

At more terrible, some datasets basically need adequate metadata to permit the information to be appropriately deciphered. For model, the mains wiring interfacing meters to one another and to machines frames a tree (with the entire house meter at the root and apparatuses at the leaves). This tree structure has more than two levels in some datasets yet the metadata once in a while determines the wiring tree. An auxiliary necessity is for information depicting the conduct of apparatuses (for example a likelihood dispersion portraying the normal times each day that every apparatus is utilized). This earlier information can be utilized to adjust the assessments delivered by a disaggregation framework. Such information is accessible in research papers and industry reports, yet not in a machine-discernible structure. At last, buyers are probably not going to invest energy into preparing a disaggregation instrument. Accordingly, if open-source disaggregation arrangements, for example, are to be suitable as purchaser confronting disaggregation arrangements then scientists must appropriate pre-prepared models for every apparatus. In the event that these models hold fast to a standard metadata pattern, at that point various programming frameworks can trade models. Against this foundation, we propose a various leveled metadata construction for vitality disaggregation.

In particular, our diagram models power meters, machines (counting earlier information such as likelihood conveyances portraying common occasions of utilization and boundaries depicting construed models of apparatuses), structures and datasets. Despite the fact that we have bent over backward to guarantee that our proposed composition and controlled vocabularies catch the data present in all the datasets we know about, our blueprint can without a doubt be improved thus the construction is introduced as an open-source project1 (under a tolerant Apache 2.0 permit) to which commitments are generally welcome!

II. RELATED WORK:

Think about two qualities, u and v, in the database, where v = F(u). To keep up the consistency of information, at whatever point u changes, F should be affected to re-register v and update its incentive in the database. This is direct for the situation where F could effected by the DBMS, e.g., SQL or C work. In this case, we address the all the more testing situation where F is a human activity, e.g., directing a wet-lab try, taking manual estimations, or gathering device interpretations. For this situation, when u changes, v stays invalid (conflicting with present estimation of u until the human activity engaged with the deduction is performed and its yield result is reproduced into the respective database.



Dependency function	Type	Description	Source(s)	Dest.	Predicate(s)
F1	Computable	Subtract one from the source	T.t2	T.t1	None
F2	Real-world	Wet-lab experiment	T.t2, T.t3	T.t4	None
F3	Computable	Multiply the source by 2	T.t4	T.t5	None
F4	Real-world	Wet-lab experiment	T.t5	S.s1	S.T_fk = T.T_pk
F5	computable	Sum the sources	S.s1, S.s2	S.s3	None
F6 (in Op# 7)	Real-world	Wet-lab experiment	S.s1	S.s3	None

Op.#	Operation	Triggering event	Consequence	Database state after the operation
1	Update cell (t2,r1) from 10 to 4	---	-Update cell (t1,r1) with value 3 -Invalidate cell (t4,r1) -Invalidate cell (t5, r1) -Invalidate cell (s1,r1) -Invalidate cell (s3,r1)	T: r1 1 3 4 5 6 12; r2 2 1 2 7 4 8 S: r1 100 70 80 150 1; r2 200 30 40 70 2
2	Update cell (t4,r1) from 6 to 13	Activity *F2* is conducted using the current values of cells (t2,r1) and (t3,r1). The output value is 13	-Validate cell (t4, r1) -Update cell (t5,r1) with value 26 -Validate cell (t5,r1)	T: r1 1 3 4 5 13 26; r2 2 1 2 7 4 8 S: r1 100 70 80 150 1; r2 200 30 40 70 2
3	Update cell (s2,r1) from 80 to 20	---	-Update cell (s3,r1) with value 90	T: r1 1 3 4 5 13 26; r2 2 1 2 7 4 8 S: r1 100 70 20 90 1; r2 200 30 40 70 2
4	Insert into T tuple r3	---	---	T: r1 1 3 4 5 13 26; r2 2 1 2 7 4 8; r3 3 8 9 4 7 14 S: r1 100 70 20 90 1; r2 200 30 40 70 2
5	Update cell (t3,r3) from 4 to 8	---	-Invalidate cell (t4,r3) -Invalidate cell (t5, r3)	T: r1 1 3 4 5 13 26; r2 2 1 2 7 4 8; r3 3 8 9 8 7 14 S: r1 100 70 20 90 1; r2 200 30 40 70 2
6	Insert into S tuple r3	---	-Invalidate cell (s1,r3) -Invalidate cell (s3,r3)	T: r1 1 3 4 5 13 26; r2 2 1 2 7 4 8; r3 3 8 9 8 7 14 S: r1 100 70 20 90 1; r2 200 30 40 70 2; r3 300 3 13 16 3
7	Alter Table S Add Dependency Using F6 Source S.s1 Destination S.s3 Invalidate destination;	The derivation mechanism of Column S.s3 is modified.	-For each database cell c in S.s3 -Invalidate (c) -For each database cell c in S.s3 -The new dependency overrides the old one	T: r1 1 3 4 5 13 26; r2 2 1 2 7 4 8; r3 3 8 9 8 7 14 S: r1 100 70 20 90 1; r2 200 30 40 70 2; r3 300 3 13 16 3

Figure 1 Example operations for set of user-distinct dependencies. (a) User-distinct dependencies for model table

One author, Propose HandsOn DB, a model database motor for overseeing conditions that include human activities while keeping up the consistency of the determined information. HandsOn DB incorporates the accompanying highlights: (1) semantics and language structure for interfaces where clients can enroll human exercises into database and execute the conditions among the information things on these exercises; (2) systems for negating ,re-validating determined information; and (3) new administrator semantics that ready clients when the returned inquiry results contain possibly in-valid information, and empower assessing inquiries on either legitimate information just, or both substantial and conceivably invalid information. Execution results are introduced that review the overheads related with these highlights and exhibit the possibility and reasonableness in acknowledging HandsOn DB [3].

We have built up a framework, called INSPIRED – Canny System for a Personalized Cancer Diet to improve the wellbeing and personal satisfaction for large patients with malignant growth. It is intended for observing, cautioning, helping and assessment to remember patients just as capacity, access and recovery of clinical and individual information for building up treatment. Our framework gives clients (patients or doctors) with customized dietary plans, restoration and way of life data, all customized to each person's needs. Fat malignancy patients are educated regarding, and taught about, the job of nourishment as a first and best safeguard in improving survivability and personal satisfaction. The proposed framework investigates quiet information, for example, essential signs (weight,

tallness, Body Mass Index BMI, circulatory strain), physical action, sort of malignant growth, treatment, symptoms and utilizations the patient's profile to construct a custom eating routine that depends on complete consideration arranging.

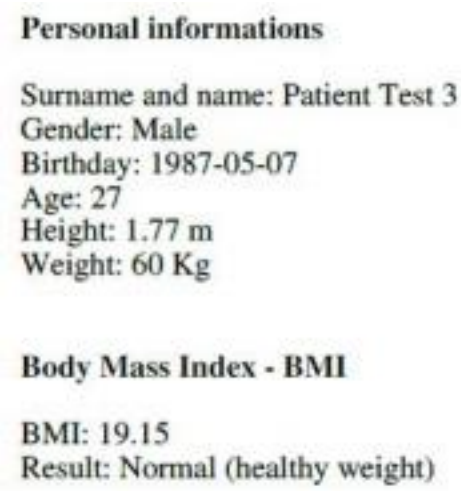


Figure 1 : BMI Report

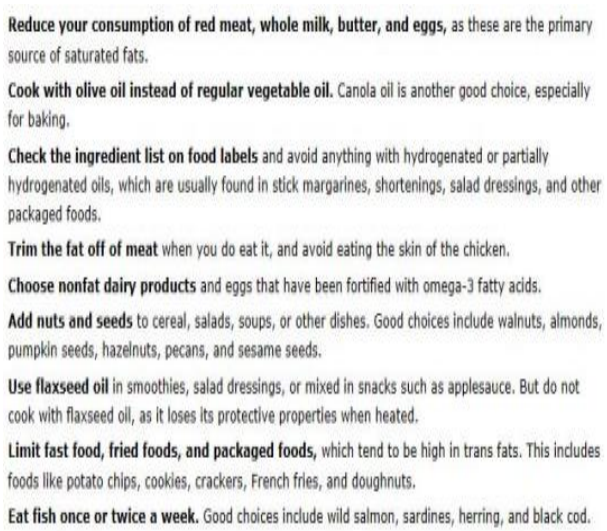


Figure 2 INSPIRED – A Personalized Diet Interface: And High-fiber, cancer-fighting foods

The Cancer Nutrition Personalized Tips System depends on My SQL (Structured Query Language) design that is client amicable and can be extended to future needs.

To display support for human learning, rules (for example activating occasion conditions-activities) can be ordered to envelop any condition of understudy learning movement enroute to fitting learning material forecast. In a specialist-based framework, each part of a versatile multiagent framework can be spoken to as operators having singular self-governance and obligation to figure it out the general objective of the framework. We also presented a context of expanded work on multiagent based Pre-appraisal System in which a demonstrating specialist utilizes the method of One v Every one of Multiple Classification rules for compute expectations for learning materials after some essential appraisal realities to an ideal idea or subject are conveyed by the pre-appraisal specialist. Utilizing SQL cosmology tree structure as the area of modelling content information, And modelling calculation is portrayed as a procedure for evaluating the all-out number of ordered standards required for the pre-appraisal framework.

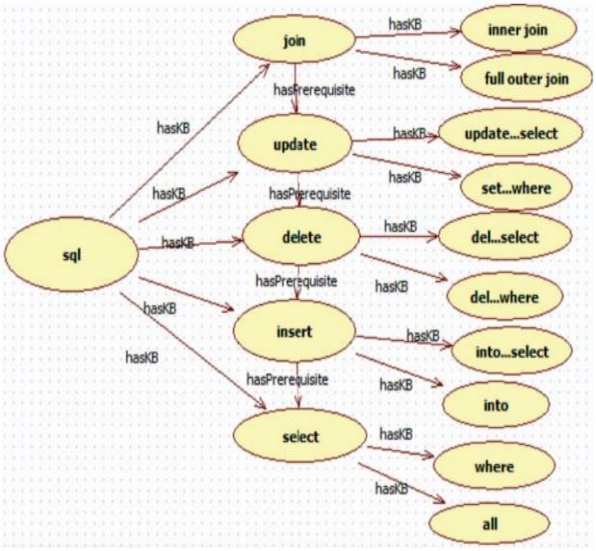


Figure 3 . Class and Subclass Relationship in SQL Regular Ontology[4]

This gauge is demonstrated to be needy on: 1) two parallel state esteems, 2) the quantity of leaf-hubs in the metaphysics tree, and 3) the quantity of essential concept(s) to an ideal idea. Also, is the learning calculation which a demonstrating operator can addition or decrement its ordered number of regulations.

we discussed earlier, we used a UK's one of the best and accurate datasets and which has more than

4 crore records. And it's means managed from 2012 and we took 3 years of data till2015 which can be publicly available in EDC website. As a part of finding human activity modeling, we been taking a reference of 19 appliance working patterns in House and managed to store in Hadoop framework after performing necessary preprocessing stages like noise removal and data modeling like converting or making the original data as more suitable for performing further data operations. And all these preprocessing has been done with python advanced data analysis libraries and after that we stored in Hadoop HDFS, very popular huge data storage, which also can support additional features like fault tolerance, and high availability. And then we wrote multiple map reduce programs to achieve the goal of modeling activities and human co-relations. And all set up is worked with the Ubuntu operating system of 14.04 version.And all the results we achieved were able to relate to human co-relations and activity modelling with high-speed data operations in high quality of structures. We achieved the processing of 4 lakh records in less than 2.5 sec in our experiments and it's a great advantage for us. We at that point quickly talk about various models that we assessed for various setting spaces. We have majorly explained the entire process in 5 steps, Including Pre-processing & Input Data Modelling, Storing Data into HDFS & Preparing data for processing, Applying the Hash Partitioning for time & day, Estimating Appliance Usage using Multiple Outputs class, Building Appliance Co-relation.

III. METHODOLOGY:

We are labelling the activities which can be extension to our earlier research where we have been considered .dat file which indicates the actual data file of UKDALE and we converted into tsv file for easy processing and converted the standard timestamp into standard date and time formats along with reducing the dataset from 6-sec resolution of data to 10 mins resolution of input data using above pandas library. This is we applied for all the devices in the home datasets. From there we stored in HDFS for reliable data storage and also available to MapReduce programming architecture. MapReduce can perform the high-speed data processing using similar kind of Java API. There we applied advanced hash functions to the HDFS input data and generated the outcomes of multiple groups of time and day differentiated record structures. And we also estimated the appliance usage of smart home big data using Multiple Outputs API of MapReduce, along with of building correlations between appliances or devices developed. For this continuation we are performing the activity labelling which can improve the estimating the human actions with more accurate and understanding manner.

a) **Activity Labelling:**
Here, based on the action performed by the human, we are predicting the task attributes like if the Washing machine and dryer is in on, we can make assumption that he/she is cleaning the cloths.

```
1289    TV,3,4,8,TUESDAY,RELAXING
1290    Dishwasher,21,22,10,TUESDAY,CLEANING
1291    Kettle,20,21,3,TUESDAY,PREPARING_BREAKFAST
1292    Toaster,21,21,3,TUESDAY,PREPARING_BREAKFAST
1293    Speaker,11,12,4,TUESDAY,RELAXING
1294    Kitchen Lights,20,21,4,TUESDAY,PREPARING DINNER
```

Figure 4 Labelling Activities

Like this, human activity can be labelled based on the usage of the device or appliance in the smart home.

b) **SQL integration:** Apache Hive integration, Google Big Query Integration

Apache Hive Integration: Hive is a SQL kind of integrated data storage component from Apache Software Foundation(ASF) which can be built on top of Hadoop software framework. Hive can be helpful in handling of data using sql kind of API, i.e. HQL (Hive Query Language). SQL is Structured Query Language that will directly interact with the data bases, regularly with RDBMS storages. It supports multiple popular databases like, MYSQL, ORACLE, etc.

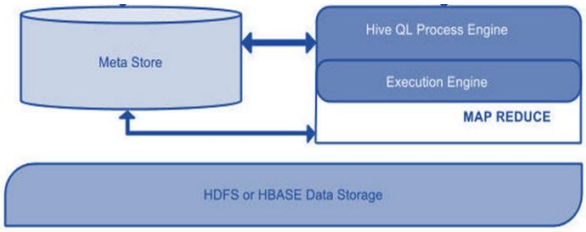


Figure 6 Hive Architecture

```
hive> SELECT day,SUM(reading) as result FROM House_Data GROUP BY day order by result;
Query ID = sureshgutur_20200730201724_ee543380-c467-44f6-ad97-06c8411f6ca0
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1595771510764_0015)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 8.43 s
--------------------------------------------------------------------------------
OK
Friday  59654692
Monday  64365760
Thursday        66895953
Tuesday 68738727
Wednesday       70245924
Saturday        70886359
Sunday  73307818
Time taken: 9.193 seconds, Fetched: 7 row(s)
```

Figure 7 Day vs Total reading output from Hive panel

```
hive> SELECT timeslot,SUM(reading) as result FROM House_Data GROUP BY timeslot order by result;

Query ID = sureshgutur_20200730201844_db4ba72b-2c24-47ff-ae1d-d635e705e302
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1595771510764_0015)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
Reducer 3 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 03/03  [==========================>>] 100%  ELAPSED TIME: 10.71 s
--------------------------------------------------------------------------------
OK
AFTERNOON       76270206
MORNING 126086106
NIGHT   132335962
EVENING 139402959
Time taken: 11.414 seconds, Fetched: 4 row(s)
```

Figure 8 Timeslot vs reading data

```
hive> SELECT deviceid,SUM(reading) as result FROM House_Data GROUP BY deviceid order by resul
t;
Query ID = sureshgutur_20200730201450_4894089a-904e-46a7-b330-2c57cbe80d3f
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1595771510764_0015)

OK
19      346875
16      1025778
17      1595346
7       5791757
9       6885151
10      7100996
11      7767475
15      11309150
18      15075651
4       16607292
12      19513659
2       26870570
6       27863592
5       49718182
8       62833612
3       63992502
13      72422219
14      77375426
Time taken: 3.717 seconds, Fetched: 18 row(s)
```

Figure 9 Device ID vs Total Reading

```
hive> SELECT day,COUNT(*) FROM House_Data GROUP BY day;
Query ID = sureshgutur_20200730201048_2857c166-0e16-49ea-a8a7-ac8947e67b1a
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1595771510764_0015)

OK
Friday  54711
Monday  52293
Saturday        53839
Sunday  53482
Thursday        53690
Tuesday 53733
Wednesday       55521
Time taken: 2.098 seconds, Fetched: 7 row(s)
```

Figure 10 Day vs Total reading

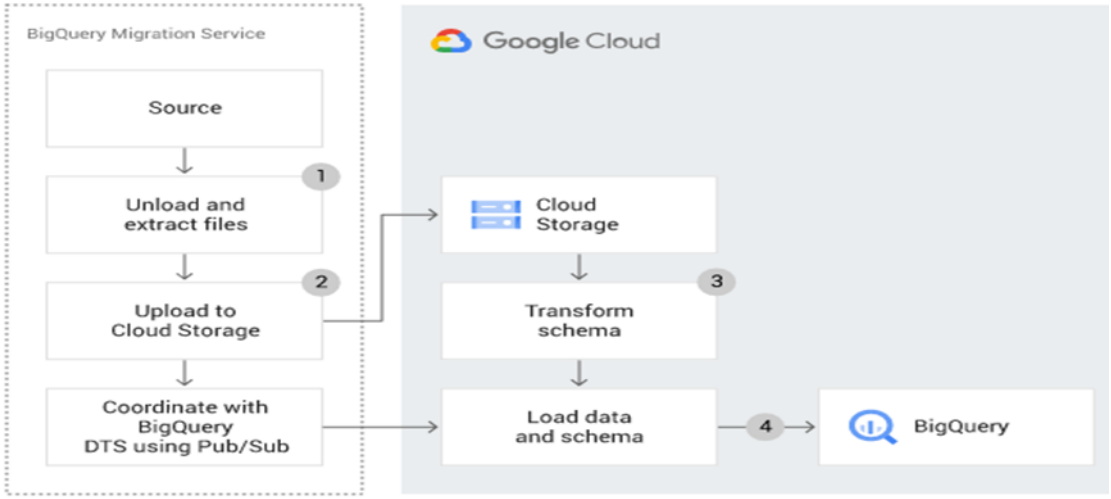


Figure 11 Google's Big Query Architecture

Google Big Query Integration: Big Query is a popular frame work for processing structure data with simple like API language SQL. It has special features like, Serverless, profoundly versatile, and financially savvy multi-cloud information distribution center and intended for business

readiness. Break down petabytes of information utilizing ANSI SQL at bursting quick speeds, with zero operational overhead is its best. It can run investigation at scale with 26%–34% lower three-year TCO than cloud information distribution center other options. It additionally can democratize bits of knowledge with a trusted, and safer stage that scales with your requirements. We will pick up experiences from information across mists with an adaptable, multi-cloud investigation arrangement.

c) Graph Representations:

As graph can make user to understand well about the insights of smart house recordings and digital readings, we want to represent our data outputs in the form of graphs and pie charts. For this we are not used any automatic libraries, and we want to make the user to experience the fresh library and developed a graph library using PHP programming language. Development of this graph library is also part of our research because client can relate easily on what actions he has to take care and what device usages he/she has to decrease. Here we been displayed the diagrams and graph representations, what we built for our analyzing disaggregated human activity and device utilizations.

IV. RESULTS & EVALUATION:

We been labelled the human activity based on energy consumption and performed the all storage/ processing operations on distributed environment for speed and efficient results. The architecture follows the distributed Hadoop framework from Apache Software foundation, and performed the operations of multiple data processing operations using MapReduce component of Hadoop. The key involvement of utilizing the energy readings and device usage can be transformed into a human understandable action and generated the activities. The performance of Hadoop framework is impressive and Hadoop has a multiple smart feature like fault tolerance and high-speed data processing. And it can store or process even tera bytes of data records.

And we executed the results with efficient and simple query language called SQL and Hive SQL for data processing. Intially we used the same apache's open source SQL component Hive and we performed various operations in easy and simple manner.

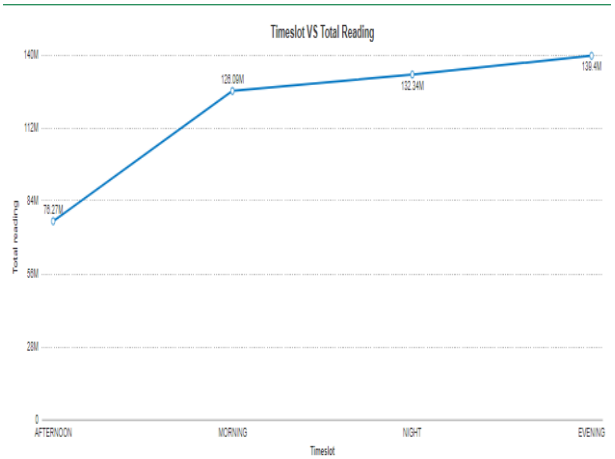


Figure 12: Line graph of Timeslot vs Reading

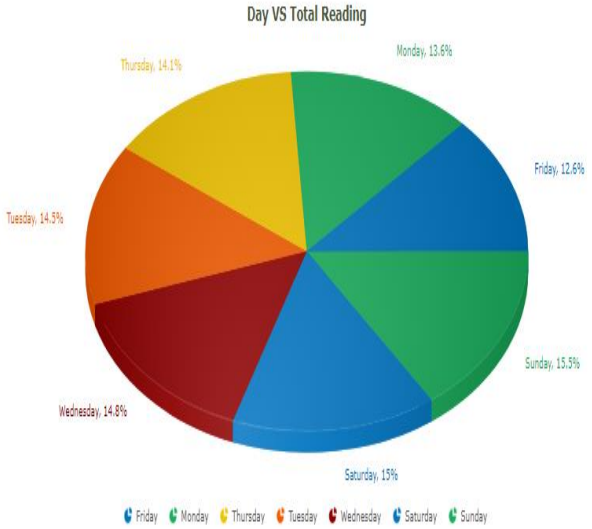


Figure 13 Day vs Total reading Pie chart

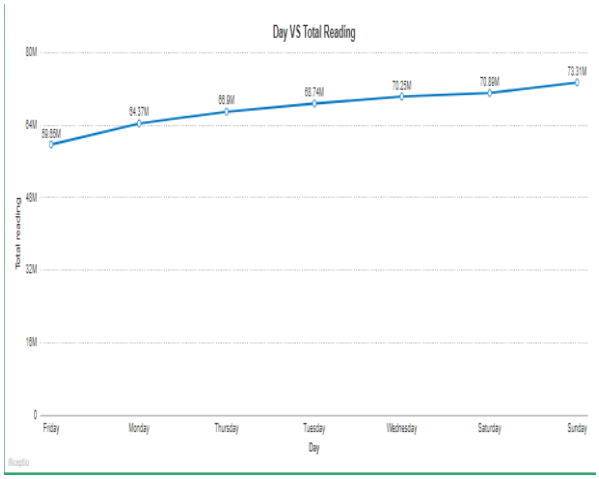


Figure 14 Day vs reading line graph

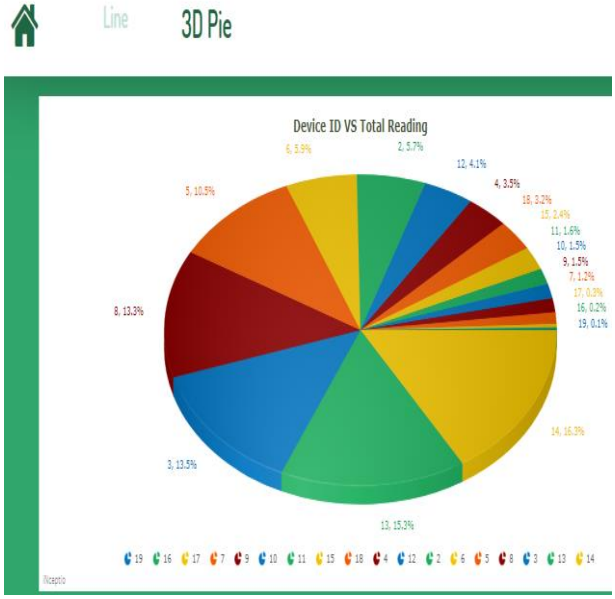


Figure 15 Device id vs reading pie graph

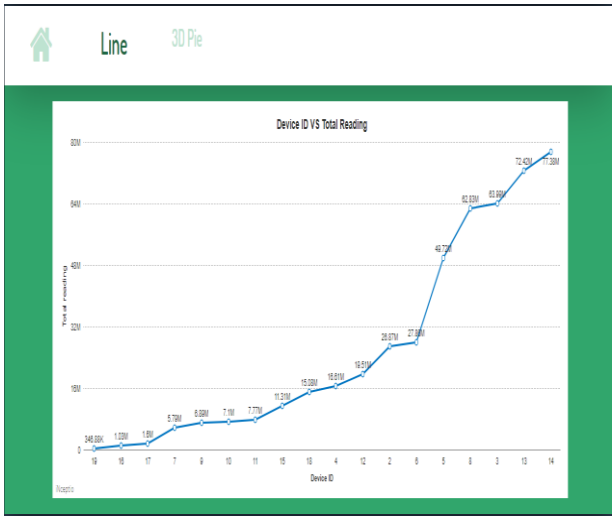


Figure 16 Device id vs Reading

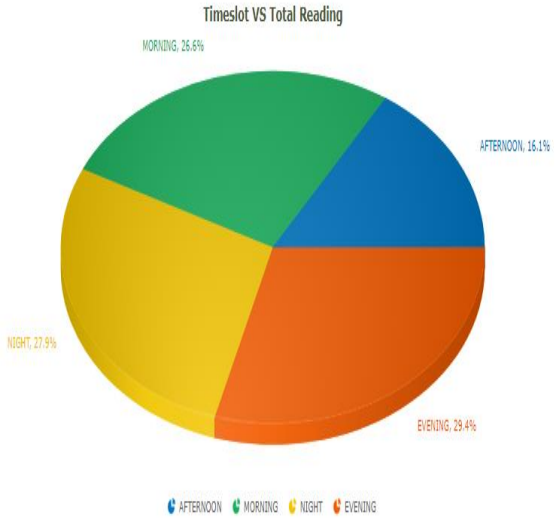


Figure 17 Timeslot vs Reading pie diagram

Then we also tested Google's BigQuery framework tool to perform various processing operations. In our test results both Hive and BigQuery has performed well and in some areas the big query has performed well and much speedy results has been generated.

V. CONCLUSION & FUTURE WORK:

We been implemented a model for activity labelling based on human actions. And enhanced our earlier research model with interfacing the SQL models like Apache Hive and Google Big Query for easy processing with simple queries and making processing and analysing to reach more users with less efforts along with development of effective pictorial representations of resultant data with graph models built by PHP libraries. However, we built and researched to reduce the usage of energy consumption, we been analysed and provided the graph representation of data usage. And the results were efficient and can be used to track the device usage and human activities especially we are dealing with special cases where the household can't monitor all these things. And any disabled or patient behaviour tracking. In future we want continue the monitoring house hold activities and want provide the best insights and enriching the advantages of energy consumption and other activity tracking with high end technology with reliable and efficient storage and

processing. Even the graph we built in PHP also got high usability and easiness.

REFERENCES

[1] Singh, Shailendra, and Abdulsalam Yassine. "Mining Energy Consumption Behavior Patterns for Households in Smart Grid." IEEE Transactions on Emerging Topics in Computing 7, no. 3 (July 2019): 404–19. https://doi.org/10.1109/TETC.2017.2692098.

[2] Kolter, J.Z.; Johnson, M.J. REDD : A Public Data Set for Energy Disaggregation Research. Proceedings of the SustKDD Workshop on Data Mining Applications in Sustainability, San Diego, CA, USA, August 2011; pp. 1–6.

[3] M. Eltabakh, W. Aref, A. Elmagarmid and M. Ouzzani, "HandsOn DB: Managing Data Dependencies Involving Human Actions," in IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 9, pp. 2193-2206, Sept. 2014, doi: 10.1109/TKDE.2013.117.

[4] K. E. Ehimwenma, M. Beer and P. Crowther, "Student modelling and classification rules learning for educational resource prediction in a multiagent system," 2015 7th Computer Science and Electronic Engineering Conference (CEEC), Colchester, 2015, pp. 59-64, doi: 10.1109/CEEC.2015.7332700.

[5] O. Geman et al., "Intelligent System for a Personalized Diet of obese patients with Cancer," 2014 International Conference and Exposition on Electrical and Power Engineering (EPE), Iasi, 2014, pp. 528-531, doi: 10.1109/ICEPE.2014.6969964.

[6] M. Idris, S. Hussain, M. Ahmad and S. Lee, "Big Data service engine (BISE): Integration of Big Data technologies for human centric wellness data," 2015 International Conference on Big Data and Smart Computing (BIGCOMP), Jeju, 2015, pp. 244-248, doi: 10.1109/35021BIGCOMP.2015.7072838.

[7] https://phppot.com/php/creating-dynamic-data-graph-using-php-and-chart-js/

[8] https://cloud.google.com/bigquery

[9] https://cwiki.apache.org/confluence/display/Hive/Tutorial

AUTHOR DETAILS:

Ravi Kanth Motupalli, a Research Scholar of Computer Science & Engineering in Acharya Nagrajuna University, Nagarjuna nagar. He is having 12 years of teaching and research experience. He completed his B.tech under JNTU and M.tech under JNTUK. He is engaged in **" Behaviour analytics on disaggregated smart home using Big Data**" Project developing process**.**

Prof. O.Nagaraju received Ph.D(CSE) and M.E (C.S.E) from Acharya Nagrajuna University , Nagarjuna nagar. He is having 13 years of Teaching Experience in various Engineering Colleges. At present he is working as Asst.Professor, Department of computer science, Government Degree College, Macherla-522426.