Volume 2, Issue 5 May 2016 ISSN NO: 2454 - 7514

North Asian International Research Journal Consortium

North Asian International Research Journal

Of

Science, Engineering and Information Technology

Chief Editor
Dr. Bilal Ahmad Malik

Publisher

Dr. Bilal Ahmad Malik

Associate Editor

Dr.Nagendra Mani Trapathi

Honorary
Dr.Ashak Hussain Malik

IRJIF IMPACT FACTOR: 3.01

Welcome to NAIRJC

ISSN NO: 2454 -7514

North Asian International Research Journal of Science, Engineering & Information Technology is a research journal, published monthly in English, Hindi, Urdu all research papers submitted to the journal will be double-blind peer reviewed referred by members of the editorial board. Readers will include investigator in Universities, Research Institutes Government and Industry with research interest in the general subjects

Editorial Board

M.C.P. Singh Head Information Technology Dr C.V. Rama University	S.P. Singh Department of Botany B.H.U. Varanasi.	A. K. M. Abdul Hakim Dept. of Materials and Metallurgical Engineering, BUET, Dhaka
Abdullah Khan Department of Chemical Engineering & Technology University of the Punjab	Vinay Kumar Department of Physics Shri Mata Vaishno Devi University Jammu	Rajpal Choudhary Dept. Govt. Engg. College Bikaner Rajasthan
Zia ur Rehman Department of Pharmacy PCTE Institute of Pharmacy Ludhiana, Punjab	Rani Devi Department of Physics University of Jammu	Moinuddin Khan Dept. of Botany SinghaniyaUniversity Rajasthan.
Manish Mishra Dept. of Engg, United College Ald.UPTU Lucknow	Ishfaq Hussain Dept. of Computer Science IUST, Kashmir	Ravi Kumar Pandey Director, H.I.M.T, Allahabad
Tihar Pandit Dept. of Environmental Science, University of Kashmir.	Abd El-Aleem Saad Soliman Desoky Dept of Plant Protection, Faculty of Agriculture, Sohag University, Egypt	M.N. Singh Director School of Science UPRTOU Allahabad
Mushtaq Ahmad Dept.of Mathematics Central University of Kashmir	Nisar Hussain Dept. of Medicine A.I. Medical College (U.P) Kanpur University	M.Abdur Razzak Dept. of Electrical & Electronic Engg. I.U Bangladesh

Address: - Dr. Ashak Hussain Malik House No. 221 Gangoo, Pulwama, Jammu and Kashmir, India - 192301, Cell: 09086405302, 09906662570, Ph. No: 01933-212815,
Email: nairjc5@gmail.com, nairjc@nairjc.com, info@nairjc.com Website: www.nairjc.com

IRJIF IMPACT FACTOR: 3.01

CARPOOLING APPLICATION FOR SMART CITY USING NEAR NEIGHBOR JOIN

PRIYANKA GHORPADE [1], DHANSHRI PATIL [2], DIPTI BHUNDE [3], & POOJA MAHAJAN [4]

Department of Computer Engineering,
Nutan Maharashtra Institute of Engineering and Technology

***Abstract*—The work proposed nearest neighbor join i.e. similarity join whose goal is to find functions whose having almost similar properties. Instead of giving exact results it will give approximate results on the basis of the mapping and characteristics of the data. And it can do so on the large scale due to join function. It uses Scalable Approximate Join (SAJ), for this large amount of joining of billions of objects. The analysis shows that the SAJ is scalable and generates good results.**

By using this approach the user proposed a new smart travel guide for the user that will provide the information of tourism with value added services. It helps user to find out its destination along with the other user by using the near neighbor join method. It take help of the GPS to find out the users location as well as the destination too.

***Index Terms*— Android, Similarity join, Map-Reduce, Top-Down Pipeline, Bottom Up Pipeline**

I. INTRODUCTION

Now a day mobile phone is a necessary half of the people's life. There's unendingly rising during a variety of mobile computing applications, focused on the people's daily life. In such applications, location dependent systems are detected as a vital application. Such application that presents the design and implementation of such a location is usually renowned as Carpooling Application for good town. It proposes design of mobile for Carpooling Application for automaton Mobile Phones that is in a position to give motion info to the mobile users handily. This system takes advantage of light-weighted mash up technology that will mix a lot of than one information sources to produce added services, whereas overcomes the restrictions of mobile devices.

The Carpooling application is meant to individual's travel, to create it cash saving and time saving too. The applying can permit the user to pick his/her role i. e. driver or rider. This may result in the user to login to the applying.

The rider and therefore the driver can enter his location, destination and time of travel. The rider gets info concerning the drivers United Nations agency locomote the route. The rider will make sure a specific driver and a message are sent to the motive force, United Nations agency additionally confirms the rider and consequently the seats are reduced.

II. GROWING IMPORTANCE OF ANDROID MOBILES

[Maintain gps related data.]

Android is software for the mobile devices that includes an operating system, middleware and key applications. The Android SDK provides the tools and APIs necessary for developing applications on the Android platform using the Java programming language. The Android is a Linux-based operating system for mobile devices such as Smartphone's and tablet computers. It is developed by the Open Handset alliance led by Google.

It is the largest community of developers writing applications that extend the functionality of the device. Developers write primarily in a customized version of Java. The Android applications can be downloaded from the third-party sites or through online store such as Google Play which is maintained by Google.

III. IMPLEMENTATION

The advancements in the field of technology are being increasing day by day. Few years before a new technology called OLA cab was introduced where user download the Car Booking application, users can do the same booking process as it was in the booking facility. But in this application only one user can book only one cab. And the share was paid by the single user only.

In this carpooling application one cab is assigned to the more than 3 users who are sharing one route. So that the road traffic can also be reduced and the fuel consumption is also reduced. Also the female user can select her co-passenger and the route with great priority.

For this application the paper proposed two algorithms,

1. Near Neighbor Join
2. Map Reduce

1. Near Neighbor Join:

The algorithm worked with two approaches as follows:

i. Bottom-Up Approach:

Require: I, the input dataset; N, the estimated size of I; k; n; m; F,P ,FJ

1: Set<Object> input = Open (I);

2: Set<SajObj> current = Convert (input);

3: level = 0, N'=N

4: while N' > n do

5: Map Stream<String, SajObj> shuffled = Shuffle (current, N', n, F, P)

6: Map<String, SajObj> clustered = Partition Cluster (shuffled, m, k, Fj));

7: current = Materialize (clustered, level);

8: level++, N'=N' / (n/m)

end while

Return: current, the final set of representative objects with size <= n; level, the final level at which BU phase stops,

ii. Top-Down Approach:

- Require: BU0,BU1,.....Buf the level-by-level objects of type SajObj produced by the BU phase; P; p; F// we first declare some variables used in each iteration

1: Map<String, SajObj> from cands, to cands; // candidate objects from different partitions to be compared.

2: Set<Pair>bu pairs, td pairs; // pairs already computed by BU or being computed by TD. // get the top-p pairs from the final BU level

3: Set<Pair> top = In Memory Top P (Buf, P, Fj, Fr);

4: for level = (f-1) -> 0 do

5: (from cands, to cands, bu pairs) = Generate Candidates (Bulevel, TOP) // define type Set<SajObj> as SSO for convenience

6: Map<String,(SSO, SSO)> grouped = Join (from cands, to cands); // Compare And Write writes to TD

7: td pairs = Compare And Write (grouped, Fj, level);

8: Set<Pair>all_pairs = bu_pairs U td_pairs;

9: if level > 0 then

10: top = Distributed Top P (all pairs, P, p, F

11: end if

12: end for

13: Return: TD0,TD1,.....,TDf, sets of all SajObj objects that are materialised at each TD level.

IV. FEASIBILITY CHECK

Place data in terms of dataset and their utilization.

1. Economical feasibility :

There is no cost involve in successful execution of this software since it is self contained, standalone program. And it doesn't require any software to process on it. Additional hardware support other than the conventional hardware is not required. Due to case of operation of the system no need of trained programmer is required.

2. Technical feasibility:

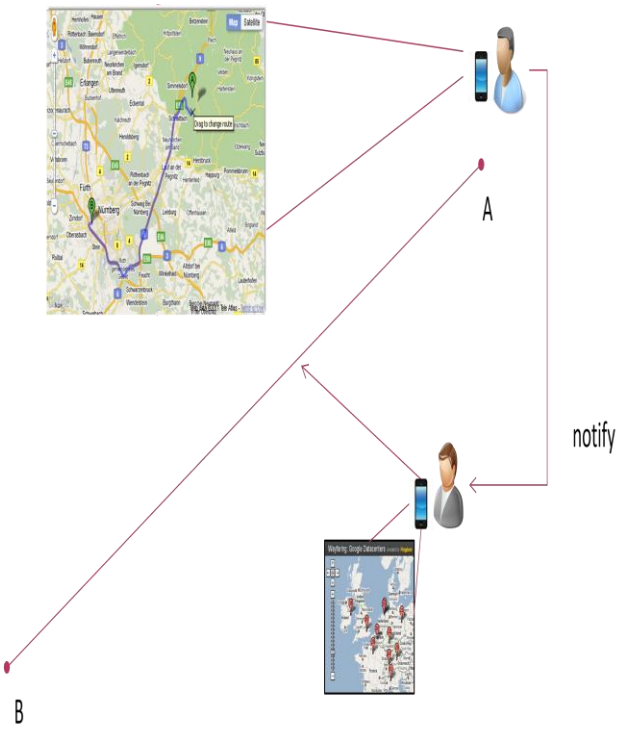
The project work can be done with the current equipment existing software and available personnel. The proposed system needs no new

equipments installed at the users end. Technically this is very much safe and sound.

3. Operational feasibility

Considering an average user's point of view, we have maintained app design as easy to handle as possible. User can understand and interact the functionality of app in short span of time and if required, Help option is given.

System design



The system is made up of 2 components:

1. Application website
2. User android application

Application website

This system provides website for user. All features given in android app can also be used on website. And if required, app can be downloaded from website.

The User Application

Using the user application, the user (passenger) can do the following tasks

1. Signup and Login

This is the first procedure to know the user's information. User need to register before using this application. During the registration user must fill his personal information like name, address, email, mobile number and set a username and password for his account.

The next time whenever user wishes to search for another user on same route or post new travel, user can simply login into his account using username and password which he has registered. All these information are stored and can be accessed from the cloud server.

2. browse

i. User can search for other user's which are going to user's desired destination. User puts name of destination. An automatic googleapi is working in background which suggests destinations available all over the globe.

ii. A asynchronous task runs when user searches from destination. Task searches for the entries in database which are for that destination.

iii. A list is generated based on single input (Destination). When user clicks item in list, information of the user and its travelling route is displayed. User can book a seat from this screen onwards.

iv. Advanced search feature is given in search screen. This feature encompasses second algorithm in our project named "near neighbour join". We give user multiple options to choose from. Based on these criteria, a function is invoked which checks for parameter and minimizes result based on parameter passed. Thus we accomplish our algorithmic implementation of finding near neighbours (nodes / users that matches with current given criteria)

3. Post New

User can post new carpool travel information from this section. He/She has options of Travel source, Destination, Day, Time, Via, Travel visibility. (Goes to Travel master). In this section we implemented Map Reduce functionality described in paper.

We have information of location of each user in tracker_master table. So we made a function which will do following

1. Fetch current location of user
2. Send that info to server which has one calculation function.
3. Function calculates distance between user and his/her friend.
4. A temporary view is created in database which has Distance mentioned.
5. A threshold distance value is taken from user (current app have default 5 value).
6. A list of users is generated which satisfies less than or equal to threshold value. And this will be used for sending our travel information.
7. So by this way we implemented map reduce functionality as we map each user's location periodically and then perform our calculations to fetch nearby user's so that notification can be sent to them only (not our).

4. Surrounding Info

User can search for surrounding amenities and establishments from

a. Atm
b. Bank
c. bus_station
d. train_station
e. cafe|food|restaurant
f. establishment
g. hospital
h. parking
i. gas_station

5. Emergency Help

Emergency help is a novel functionality in our app. This functionality chooses app user's friends in user's surrounding dynamically and send alerts to them in case of emergency. In existing system, apps tend to use static data provided to application in such cases but our app makes selection of guardians on dynamic basis based on the distance.

V. SUMMARY

This paper gives the idea of a mobile cab pooling application on android phone. This application developed by using smart phones which can change the way of people to rent a cab in future. This application can be modified according to any type of transport system. This android application to rent a cab through an android mobile.

This android application reduces the manual work of both user and administrator. This app also reduces the road traffic and saves fuel also.

VI. FUTURE WORK

Carpooling system is extremely effective means that to cut back pollution and therefore the congestion of vehicles in cities. It also provides associate degree eco-friendly thanks to travel. It additionally provides an opportunity to fulfil new individuals. As these days most people like personal vehicle to travel owing to delay caused publicly transport system and luxuries provided by personal vehicles. Pre-registration ensures security, as solely known individuals get into the vehicle so that trust will be established. The individuals registered can be assigned specific days on that they ought to take their personal vehicle, so no inconvenience is caused to its registered passengers for daily commute.

Thus the planned carpooling systems are effective in reducing setting pollution.

This work provides a symbol of conception of the automotive Pooling application. There's scope of improvement by applying many measures available. The future work with regards to the current paper is summarized as follows:

• Currently, System uses GPS triangulation system which is very hefty on normal user devices which almost drains battery in short period of time. So there's surely scope of improvement in existing system.

VII. ACKNOWLEDGEMENTS:

WE ARE THANKFUL TO OUR PROJECT GUIDE PROF. DHANSHRI PATIL AND PROF. ASHWINI JADHAV FOR THEIR SUPPORT. ALSO ALL THE STAFF OF COMPUTER DEPARTMENT FOR COORDINATION.

REFERENCES

[1] “Near neighbor join,” Kllapi, H., Harb, B. ; Cong Yu, Dept. of Inf. & Telecommun., Univ. of Athens, Athens, Greece

[2]"Processing Theta-Joins using Map Reduce", Alper Okcan, Mirek Riedewald, in SIGMOD,2011

[3] "High-dimensional kNN joins with incremental updates", Cui Yu, Rui Zhang, Yaochun Huang, HuiXiong.

[4]“Similarity join size estimation using locality sensitive hashing” Hongrae Lee, Raymond T. Ng, Kyuseok Shim

[5] "Efficient Parallel kNN Joins for Large Data in Map Reduce", Chi Zhang1 Feifei Li2 Jeffrey Jestes2

[6]" Efficient Processing of k Nearest Neighbor Joins using Map Reduce", Wei Lu Yanyan Shen Su Chen Beng Chin Ooi

[7] "Fast Approximate Nearest-Neighbor Search with k-Nearest Neighbor Graph", Kiana Hajebi, Yasin Abbasi-Yadkori, Hossein Shahbazi and Hong Zhang.

Publish Research Article

Dear Sir/Mam,

We invite unpublished Research Paper,Summary of Research Project,Theses,Books and Book Review for publication.

Address:- Dr. Ashak Hussain Malik House No-221, Gangoo Pulwama - 192301 Jammu & Kashmir, India
Cell: 09086405302, 09906662570,
Ph No: 01933212815
Email:- nairjc5@gmail.com, nairjc@nairjc.com , info@nairjc.com
Website: www.nairjc.com