Intelligent System for Searching Nearest Services using Multi-Agent Approach

Waqar Mirza M.[1], Ana María Martinez-Enriquez[2], Aslam Muhammad[1], Afraz Z. Syed[1], Arslan Lodhi[3]

[1] Department of CS & E, UET , Lahore, Pakistan
[2] Department of Computer Science, CINVESTAV-IPN, D.F., Mexico
[3] Department of Information Technology, APPSTER Tech Ltd., Lahore, Pakistan

mirzamwaqar@hotmail.com,{maslam,afrazsyed}@uet.edu.pk,
ammartin@cinvestav.mx,arslan.lo@eleget.net

Abstract. In this paper, we present the multi agent approach for developing an intelligent system for mobile users to find most of the different type of deals/services available nearby to them at unknown places. Now days, smart phone have the ability to determine the exact position of a terminal on the earth. By use of the ability the system collects the user's position and discovers the services available near to him. Many services are being offered by mobile companies and Web sites to look for locations, finding routs, and tracking devices. However, the Web content is updated after certain interval of time and is not so much interactive. For instance, the information like doctors on duty, available medicines, and beds is outdated and cause the wastage of time as well as money of visitors. We resolve this issue by means of design and development of an intelligent system whose architecture is based on multi-agents who collaborate and cooperate with each other to present the most accessible and useful services available in an interactive way. The results obtained from the experiments are very encouraging and much appreciated by the users of this system.

Keywords: Multi-agent system, GPS, location based services, Android application.

1 Introduction

Through the advent of multi agent technologies and mobile phone applications, the organizations now have many options for determining where the computing resources of an application take place [1]. Bottleneck in the overall IT infrastructure can be overcome by use of these technologies. The rapid advancements in the area of Web and Mobile applications have encouraged the provision of GPS (Global Positioning Systems) [2] based mobile application. The GPS is a system based on almost 24 satellites placed into an orbit which navigate to make up a network. GPS is available

anytime, anywhere and any weather condition and also to anyone without any charges. If the user have GPS receiver in his phone then he can easily determine his position on earth. It tells the exact position of the user when he is in the range of at least three satellites of GPS network [13]. The GPS express the position in terms of latitude and longitude values.

Finding of exact location of a user is the basic of location based services (LBS). As an example, when a user wants to search any restaurant or shopping mall near to him using any mobile application, it is necessary to first determine the current location of the user and then discover all the nearest services to that position. The GPS and LBS play a vital role in mobile application to facilitate its user. It is a challenging task to find the nearest resource from a user's current location when he is travelling to an unknown location. When a user is at strange place, it is important to identify nearby services (fuel station, restaurant, vehicle service center etc.) to the needy people. This reduces the amount of stress a person feels during his journey and also save his time.

Wide range of LBS are available that can helps the user to finding a service or inform the user about his current location and ways to destination [14]. One of them is a navigation service, uses the digital map along with the labels of popular streets and buildings, use to find the direction during journey from current location to destination. Google Maps [18] is an example of this type of service where user can see different routes to reach a specific destination along with their distances and estimate times. When a digital map is extended with the facility to find the physical services then it is called yellow page service. In further extension, when a system is able to locate the user's current position and determine the nearby object and resources then these types of services is called location aware information services. However, all these services do not provide timely updated information, for instance, a tourist at any foreign place may need to visit an auto teller machine to draw money. Any location based service either offered by the bank or other vender may tell the exact site of the money cube but it is not able to inform about the case situation. If the foreigner desires to withdraw high denomination notes then the machine would not be able to answer such queries. The reason is that these systems update their values after a specific interval of time. Therefore, the problem often occurs when a user wants newest facts about any service, as he needs it urgent, and the system is unable to provide it. Likewise, cases exist where travelers or even residents have to waste a lot of time and money in terms of fuel in search the exact service as desire.

In this research, we extended the location aware information service by not only searching the nearest available service/resource but also with the current states of that service either it is presently available or not which in fact reduces the time and effort of people. We tackle this issue by the design and development of an intelligent system whose architecture is based in multi-agents. Agents can communicate, collaborate, cooperate, and act autonomously. The purpose of the system is to save the time of the user and provide comfort to him during his trip. The major objective of this research is to get a highly cost-effective application which caters the needs of the users and facilitates them in a most effective manner. The system offers many different types of businesses deals on single platform so that users can give a comparative look on multiple businesses. The purpose is to improve the existing system’s services by

increasing user satisfaction, providing the quality and creating a controlled information environment. The system offers the better performance by providing desired flexibility, fast response, ability to support changes and ability to maintain the quality of services.

The rest of the paper is organized as follows: related work is discussed in Section 2. Section 3 describes in detail the intelligent architecture of LBDSS (Location Based Deals Searching System). Experiment and their obtained results are presented in Section 4. Conclusion and future perspectives are given in Section 5.

2 Related Works

Lots of work has been done in regards to help users in stranger places when they are unknown to that location and need any services immediately. All these existing system may help their user in such a situation but sometimes they are not successful in helping them correctly. The existing system update their databases after a long time that's why they do not provide accurate information to their user and instead of helping the user to remove them from problems they create more problems for them. Here we will discuss some of them in details below.

Man Lung Yiu et al. [3] studied the aggregate nearest neighbor queries to create a function that help to find the minimum aggregate distance between the special object and special network (e.g. road). Different aggregate techniques and function are considered to exploit Euclidean distance bounds, special access methods and network distance materialization structure. In their research they presented three algorithm, IER incrementally retrieves Euclidean aggregate nearest neighbors and computes their network distances by shortest path queries until the result cannot be improved. TA and CE explore the network around the query points until the aggregate nearest neighbors are discovered. These techniques are combined with special access method and shortest path materialization technique to get the minimum distance.

S. R. Balasundaram et al. [15] uses the query management to find out the most appropriate location of required services needed by a mobile user. The major objective of the paper is to provide the information to needy people who are searching for the resources during travel at their minimum distance. The location of the user is obtained using GPS and the distance of user's future location and available resource point is calculated and the resource with minimum distance available is returned to user.

Hae Don Chon et al. have developed a location based application named NAPA (Nearest Available Parking Lot Application) [16] used to assists users to find a nearest parking space on campus. NAPA uses the feature of LBS, wireless communication and directory services (LDPA) to achieve his goal. The similar kind of idea named SIAPAS [17] is also presented by G Mendez et al. The design of SIAPAS is consist of a set of six independent modules that communicate with each other using web services. Communication module keep track of parking space, GPS gets the current position, Voice provides driving assistance, GUI for user interaction, Outside Parking Manager to control global parking, Inside Parking Manager keep

track of parking routes and Configuration module manages the GPS device configuration.

Also there are available some mobile application [4] [5], deals aggregator in nationwide, is not much up to the mark as they should be. The goal of these applications is to provide a unique location based mobile solution to aggregate hundreds of deal sites and make the process of saving money easier and less time consuming. When a user subscribes to these services he get alerts whenever he is close to a deal.

All the above discussed systems provide the limited information to their users. They only tell the user which deals/service is available near and how much far away from him. But they do not tell the user either the required item is currently available at that shop or either the shop is currently open or closed. Normally what happened when a user uses this ordinary application, he thought to select the most nearest location to get appropriate service and unfortunately when he reaches there, the shop founds to be closed or the required service/item is not currently available at that particular time or out of stock. So, instead of saving time they lost it and also the efforts. On the other side, the LBDSS is developed by keeping these short comes in mind. The system has the latest information whenever user needs it. So our key idea is to develop a platform for business dealers and customers for communication so they can easily get the information by just few taps of fingers.

3 Architecture of Intelligent LBDSS

The proposed system is developed by using multi-agent technology where multiple agents communicate and coordinate with each other to fulfil the assigned task. The foremost objective of this system is to facilitate the users in a most effective manner and provide him latest information about the deals or services. To use LBDSS, the user just need an android [10] [11] supported handset and internet connection in it. The user installs LBDSS client application in his mobile which co-ordinates with the server to get the deals according to the user's search criteria.

3.1 Agents in LBDSS

Software architecture describes the major components of the system and the relationship between them [6] [7]. The overall architecture of LBDSS is based on three agents who work for the users to get required information and facilitate its user by giving him up-to-date information when he requests. The detail of these three working agents is shown in Figure 1 and also discussed each of them works as individual and as a team.

User interface and Query Agent (UIQA). As the name describes the working of this agent is to provide an interface to end user by giving him the facility to use

application and take queries from the user. The requested parameters are then sent to the Data Management Agent (DMA) to get filtered information. The DMA process the query and send back the fetched data to UIQA. The obtained result is then displayed on the users screen for further processing. The responsibilities of this agent is to provide an easy to use interface to make queries for searching required information and communicate with the DMA to get required result and present to user.

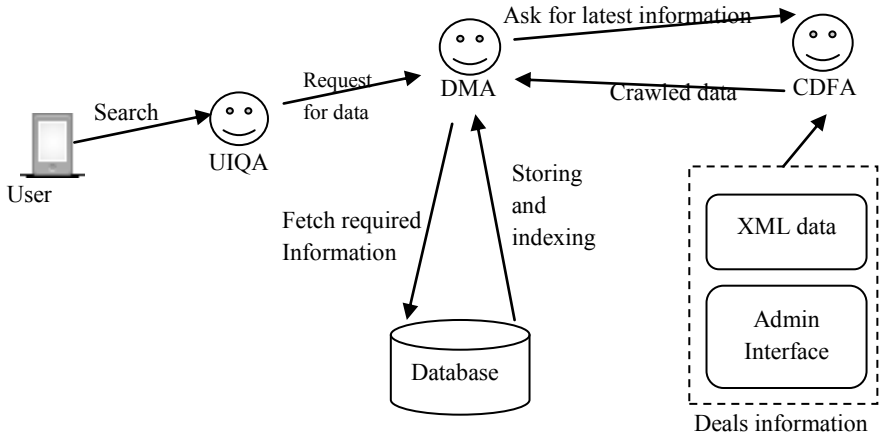

Fig. 1. General architecture of purposed system

Data Management Agent (DMA). DMA is in the middle of both agents. He got queries from the UIQA and send result back to him. The another major task done by this agent is to get crawled data from the Crawling and Data Fetching Agent (CDFA) and manage that data by making indexing of each data element and then store them in system's database. The advantage of making indexes of data element is to search them quickly and efficiently. Also if DMA did not find any up-to-date information in the system database it send request to CDFA to crawl immediately and give him latest information. The latest information means, checks services are currently available or the item user searching for is in stock or out of stock. Furthermore checks weather the shop or organization is currently open or closed. All these information is provided by the venders through an XML feed or through the HTML based interface provided to him.

Crawling and Data Fetching Agent (CDFA). The last agent CDFA is responsible of crawling the available sources and get information from them. This is a sequence based task. The CDFA visits each source one by one and send information to DMA for storing and for future using. But the most important working of CDFA is when a user requests for the information about the particular organization's deals but the LBDSS does not have up to date. Then the DMA request to CDFA to crawl that

vender's source immediately and give him the latest information. Here we have two way of getting information from the venders. One way in which an administrative interface is given to a vender where he register himself and then provide all the services and products he offers. Also it is regularly updated by the staff of the organization at short interval of time about the stock and shop status and any other information which will useful for their searchers. The second source is vender's web site. We request them to give us the update information in XML format available for CDFA on a given URL. The CDFA visits that URL after a specific interval of time and get the latest information and send it to DMA.

Communication between the agents. For communication and sharing information between the agents the system uses FIPA ACL which is encoded in XML. The same technique is used by [8] [12] in their systems for agent communication and found it very efficient, fast and easy to implement in web based environments. So considering all these things, we also decided to use FIPA ACL with XML encoding for our agent's communication. A sample agent communication message is from agent-1 to agent-2 shown below. The message is encoded in XML. It can be seen that the sender and intended recipient of the message are identified by their agent-identifiers. For the sample message, the sender and receiver agent names are AG1 and AG2, respectively. The sender and receiver agent addresses are http://ag1.masc.com:5120 and http://ag2.masc.com:5120, respectively. The actual message is enclosed in message tag. The 'type' parameter defines the type of message either it is request or reply to some request and the 'id' parameter uniquely identify each message that is used by receiver to send reply of the query for reference. The 'content' tag contains the requested value required by agent from other and arguments tag contains the parameter use to filter the required information. The following sample message is a request message send by UIQA to DMA to get the list of all restaurants near to user's current location.

```
<?xml version="1.0" ?>
<sender>
  <agent_identifier>
    <name>AG1</name>
    <address>
      <url>http://ag1.masc.com:5120</url>
    </address>
  </agent_identifier>
</sender>
<receiver>
  <agent_identifier>
    <name>AG2</name>
    <address><url>http://ag2.masc.com</url></address>
  </agent_identifier>
</receiver>
<message type="request" id="MSG-1">
```

```
  <content>Nearest  Restaurant</content>
  <arguments>
    <latitude>23.536723</latitude>
    <longitude>2.3223411</longitude>
    <current_time>2013-01-12 23:22</current_time>
  </arguments >
</message>
```

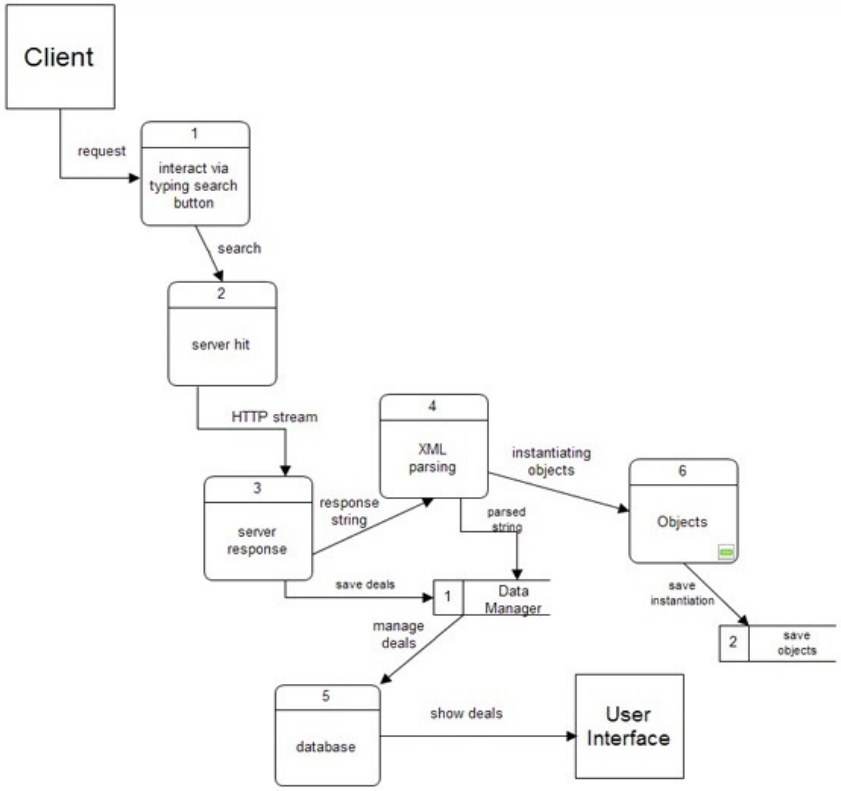

Fig. 2. Flow of information in purposed system

3.2 Methodology used for LBDSS

In this section we will discuss how actually the system works with the help of flow diagram of the system shown in figure 2. First of all user request for the required information thorough his hand held devices. For this purpose user will have to install the LBDSS client software on his mobile and mobile should have android operating system in it. The user submits his request through UIQA installed along with client application. Then UIQA send this request to server where DMA receives the request parameters and starts fetching data from available sources.

In figure 2 we can see that data from UIQA to DMA is travelled in the form of HTTP stream. Here the two agents used XML ontology [9] for communication as discussed earlier. And the request message is transferred using HTTP stream. DMA receives the request and then see in his record is the required data is in the database or not. If the data is available in system's database, then it is checked weather is it recent information or expired. If the available information is expired, the DMA request to CDFA to fetch the latest information from the vender. The vender may provide newest information through XML channel or may use the administrative interface provided by LBDSS. The latest data in XML format is sent to DMA who parse it and stored it in system's database and also sent it back to UIQA in reply to the request. UIQA takes the received information and display it to user's screen.

4 Case Studies

In order to test the performance and efficiency of the LBDSS, numbers of test experiments are performed. Initially to collect real data we visited different venders and ask them to provide data either through XML feed or use our purposed web based interface. For testing purpose we selected 30 restaurants at two major locations in the city. Then 10 end-users with android mobile, who are not much familiar with those areas, are selected and ask to visits these locations. For data inputs the android mobiles are used through which user can comfortably give the input and type the required keyword for search. User has to install the LBDSS client software on his mobile so he can get the information from the server. The LBDSS client application is easily available on our website and also on android app stores. These entire users have different brands of android mobiles.

a. search screen b. search result c. product details

Fig. 3. Screen shoots of proposed system

When user installs the software and open it in his mobile, the main screen ask him for keyword he want to search and some advance parameters if needed as shown in

figure 3(a). We asked the user to search only food items, as for testing, we used only restaurants venders. After entering the desired item in keyword box the system goes to server and request for the items which are nearest to the current location. Then the search result is fetched and displayed to its users as shown in figure 3(b). The user has the option to view the details of each product as shown in figure 3(c) and also have the choice to view the direction to that vender from his current position.

At the other end, to receive information and data from the venders we have multiple options. The first one is the XML feed. This is helpful when venders have their own web application and database. Some of the venders in our list have the XML facility and agree to give us their data in XML format. So, to get the latest information about their product and its current state, we used their XML feed which is connected with their database. The XML file is access through a URL provided by those venders to us. CDFA uses this URL to parse and get data from it. After getting the data it is sent to DMA who indexed it and store in system's database. The second option for receiving latest information from the vender who did not have the facility of XML a web based interface is provided to them where they put their data regularly to keep us up to date. The venders used this Administrative application and add/edit/update/delete their product information, their stock states and any other information that is useful by us or end user. For actual system's implementation to access this section, the vender first has to get subscription from us. A signup form is available on LBDSS web based application where vender can get registered with us and available for end user to present their latest product and services.

After using this application almost 2 hour all the user are asked to give feedback about the application and tell how it help them in searching their required item. The survey results are very encouraged and the entire users give excellent rate to this application. The major feedback we received from user is they happy to receive latest information as it stops the wastage of their time to getting their required information. It helps them in searching shops or restaurants to which they are not familiar within seconds.

5 Conclusions and Future Work

Searching an item or services in an area where you are totally new or not much familiar is very difficult task. In these cases normally the people are very confused and often the service they are looking for is near to him but due to unfamiliarity about the location they goes to wrong direction and wastes their lots of time in searching. So, the purposes system is very helpful for that kind of users. The results obtained from the experiments are very successful and very impressive. All the users recommended this system as guider which guides them while they are moving in unknown location.

The obtained results encourage us to expend this idea and make it more useful for its user. So in our future plan we decided to setup mobile alerts about the deals, the user have the facility to communicate directly with the vender to get information

through this application. Instead of ask vender to give us XML we plan to make a crawler who will visit the web pages of the vender and fetch information from them.

References

1. K. Virrantaus, J. Markkula, A. Garmash and Y. V. Terziyan: Developing GIS-Supported Location-Based Services. In: Proceedings of WGIS'2001 - First International Workshop on Web Geographical Information Systems, Kyoto, Japan, pp. 423−432 (2001)
2. T D'Roza and G Bilchev: An overview of location-based services. In: BT Technology Journal, vol. 21(1), pp. 20−27 (January 2003)
3. M. L. Yiu, N. Mamoulis, and D. Papadias: Aggregate nearest neighbor queries in road networks. In: IEEE Trans. Knowl. Data Eng., pp. 820−833 (2005)
4. http://www.blingdar.com/ (visited on 1st Jan 2013)
5. http://www.groupon.com (visited on 1st Jan 2013)
6. S. Mary: The Coming-of-Age of Software Architecture Research. In: Proc. of the 23rd International Conference on Software Engineering, pp. 656, (2001)
7. M. Nenad, S.R. David, F.R. David and E.R. Jason: Modeling software architectures in the Unified Modeling Language. In: ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 11, pp. 2−57 (2002)
8. Khaoula ADDAKIRI, Mohamed BAHAJ, Noreddine GHERABI: XML-based agent communication and migration for distributed applications in Mobile-C. In: International Journal of Modern Engineering Research (IJMER), vol. 2(1), pp. 403−407 (Jan−Feb) (2012)
9. Extensible Markup Language (XML). The World Wide Web Consortium. http://www.w3.org/XML (visited on 1st Jan) (2013)
10. http://developer.android.com (visited on 1st Jan) (2013)
11. www.designerandroid.com (visited on 1st Jan) (2013)
12. Séverac, G., Hvilshøj, M., Bøgh, S.: XML−BASED MULTI AGENT COMMUNICATION AND SCHEDULING IN INDUSTRIAL ENVIRONMENTS; In: MITIP 2010, Aalborg, (30−31 August) (2010)
13. Davies N., Cheverst K., Efrat A.: Using and Determining Location in a Context-Sensitive Tour Guide. In: IEEE Computer, vol. 34(8), pp. 35−41 (2001)
14. Virrantaus K., Markkula J., Garmash A., Terziyan V., Veijalainen J., Katanosov A., Tirri H.: Developing GIS-supported location-based services. In: Proceedings of the Second International Conference on Web Information System Engineering, vol. 2, pp. 66−75 (2001)
15. S. R. Balasundaram, Saravanarr, A.: Resource Identification Using Mobile Queries. In: ACEEE Int. J. on Communication, vol. 1(3), pp. 25 – 27 (Dec) (2010).
16. Hae Don Chon, Divyakant Agrawal and Amr El Abbadi: NAPA :Nearest Available Parking lot Application. In: IEEE Proceedings of the 18th International Conference on Data Engineering, (2002)
17. Gonzalo Mendez, Pilar Herrero, and Ramon Valladares: SIAPAS: A Case Study on the Use of a GPS-Based Parking System. In: OTM 2006 Workshops, LNCS vol. 4277, pp. 945−954 (2006)
18. http://www.google.com/mobile/maps/ (visited on 1st Jan) (2013)