

SYSTEMATIC WAY OF PROCESSING AND FETCHING DATA USING DATA MINING

Dr.G.Komarasamy , N.Balaji , J.Gokulnath , R.Hariprasath

Abstract— Big Data consists of huge modules, difficult, growing data sets with numerous and independent sources with the fast development of networking, storage of data, and the data gathering capacity. Here we are using Apriori theorem that states the characteristics data and proposes a data processing model from the data mining view for extracting useful information from these large datasets or streams of data to look for relevant or pertinent information. Here we analyze several challenges at the data, model, and system levels and using apriori theorem for clustered storage which combines multiple arrays to increase their performance, capacity and reliability. Clustering techniques is used to partition the file uploaded by the admin as a set of entities into different groups so that it will be easy for the user to download even large files efficiently.

Keywords -- Big Data, Networking, Apriori, Clustering.

I. INTRODUCTION

Data production rate has been increased dramatically and we are able store much more data than before like purchase data, social media data, mobile phone data. Businesses and customers need useful or actionable knowledge and gain insight from raw data for various purposes. It's not just searching data or databases. Data mining helps us to extract new information and uncover hidden patterns out of the stored and streaming data. The process of discovering hidden patterns in large data sets.

It utilizes methods at the intersection of artificial intelligence, machine learning, statistics, and database systems. Extracting or mining knowledge from large amounts of data, or big data. Data-driven discovery and modeling of hidden patterns in big data. Extracting implicit, previously unknown, unexpected, and potentially useful information/knowledge from data. Classification is assign data into predefined classes.

Dr.G.Komarasamy , Department of Computer Science and Engineering ,Bannari Amman Institute of Technology, Sathyamangalam, India

N.Balaji , Department of Computer Science and Engineering , Bannari Amman Institute of Technology, Sathyamangalam, India

J.Gokulnath , Department of Computer Science and Engineering , Bannari Amman Institute of Technology, Sathyamangalam, India

R.Hariprasath , Department of Computer Science and Engineering , Bannari Amman Institute of Technology, Sathyamangalam, India

Regression is predicting real value for a given data instance. Clustering is the process of grouping similar items together into some clusters. Java has been around since 1991, developed by a small team of Sun Microsystems developers in a project originally called the Green project. The intent of the project was to develop a platform-independent software technology that would be used in the consumer electronics industry. The language that the team created was originally called Oak.

Around the time the First Person project was floundering in consumer electronics, a new craze was gaining momentum in America; the craze was called "Web surfing." The World Wide Web, a name applied to the Internet's millions of linked HTML documents was suddenly becoming popular for use by the masses.

The reason for this was the introduction of a graphical Web browser called Mosaic, developed by ncSA. The browser simplified Web browsing by combining text and graphics into a single interface to eliminate the need for users to learn many confusing UNIX and DOS commands. Navigating around the Web was much easier using Mosaic.

It has only been since 1994 that Oak technology has been applied to the Web. In 1994, two Sun developers created the first version of Hot Java, and then called Web Runner, which is a graphical browser for the Web that exists today. The browser was coded entirely in the Oak language, by this time called Java. Soon after, the Java compiler was rewritten in the Java language from its original C code, thus proving that Java could be used effectively as an application language. Sun introduced Java in May 1995 at the Sun World 95 convention.

Web surfing has become an enormously popular practice among millions of computer users. Until Java, however, the content of information on the Internet has been a bland series of HTML documents. Web users are hungry for applications that are interactive, that users can execute no matter what hardware or software platform they are using, and that travel across heterogeneous networks and do not spread viruses to their computers. Java can create such applications.

Java Server Pages or JSP for short is Sun's solution for developing dynamic web sites. JSP provide excellent server side scripting support for creating database driven

web applications. JSP enable the developers to directly insert java code into jsp file, this makes the development process very simple and its maintenance also becomes very easy. JSP pages are efficient, it loads into the web servers memory on receiving the request very first time and the subsequent calls are served within a very short period of time.

In today's environment most web sites servers dynamic pages based on user request. Database is very convenient way to store the data of users and other things. JDBC provide excellent database connectivity in heterogeneous database environment. Using JSP and JDBC its very easy to develop database driven web application. Java is known for its characteristic of "write once, run anywhere." JSP pages are platform independent. You port your .jsp pages to any platform.

A JSP page services requests as a servlet. Thus, the life cycle and many of the capabilities of JSP pages (in particular the dynamic aspects) are determined by Java Servlet technology and much of the discussion in this chapter refers to functions.

When a request is mapped to a JSP page, it is handled by a special servlet that first checks whether the JSP page's servlet is older than the JSP page. If it is, it translates the JSP page into a servlet class and compiles the class. During development, one of the advantages of JSP pages over servlets is that the build process is performed automatically.

A servlet is a Java programming language class used to extend the capabilities of servers that host applications accessed via a request-response programming model. Although servlets can respond to any type of request, they are commonly used to extend the applications hosted by Web servers. Thus, it can be thought of as a Java Applet that runs on a server instead of a browser.

A Servlet is a Java class in Java EE that conforms to the Java Servlet API, a protocol by which a Java class may respond to requests. They are not tied to a specific client-server protocol, but are most often used with the HTTP protocol. Therefore, the word "Servlet" is often used in the meaning of "HTTP Servlet".

Thus, a software developer may use a servlet to add dynamic content to a Web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML. Servlets are the Java counterpart to non-Java dynamic Web content technologies such as CGI and ASP.NET. Servlets can maintain state in session variables across many server transactions by using HTTP cookies, or URL rewriting.

To deploy and run a Servlet, a Web container must be used. A Web container is essentially the component of a

Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights.

A Servlet is an object that receives a request and generates a response based on that request. The basic servlet package defines Java objects to represent servlet requests and responses, as well as objects to reflect the servlet's configuration

package `javax.servlet.http` defines HTTP-specific subclasses of the generic servlet elements, including session management objects that track multiple requests and responses between the Web server and a client. Servlets may be packaged in a WAR file as a Web application.

Servlets can be generated automatically from Java Server Pages (JSP) by the JavaServer Pages compiler. The difference between Servlets and JSP is that Servlets typically embed HTML inside Java code, while JSPs embed Java code in HTML.

The direct usage of Servlets to generate HTML (as shown in the example below) is relatively rare nowadays, the higher level MVC web framework in Java EE (JSF) still explicitly uses the Servlet technology for the low level request/response handling via the `FacesServlet`. A somewhat older usage is to use servlets in conjunction with JSPs in a pattern called Model 2, which is a flavor of the model-view-controller pattern.

Servlets belong in `WEB-INF/classes`. On this machine, the source is in Java source. `Web-Inf/classes` is the standard location for servlets and other Java classes. Resin automatically reloads and recompiles servlets, beans, and classes placed in `web-inf/classes`. You should make some changes and add errors to become familiar with Resin's recompilation and the error reporting.

MySQL ships with many command line tools, from which the main interface is 'mysql' client. Third-parties have also developed tools to manage, optimize, monitor and backup a MySQL server, some listed below. All these tools work on *NIX type operating systems, and some of them also on Windows

II. LITERATURE SURVEY

In "Impact of fairness on internet performance" by T.Bonald, 2006, it is proposed that the relevance of fairness as a design objective for congestion control mechanisms in the Internet. Specifically, we consider a backbone network shared by a dynamic number of competing flows, and study the impact of bandwidth sharing on network stability and performance. In

particular, we prove that for a broad class of fair bandwidth allocations, the total number of flows in progress remains finite as long as the load of every link is less than 1. We also show that provided the bandwidth allocation is sufficiently "fair", the backbone is virtually transparent in the sense that the throughput of the flows is mainly determined by their access rate. Neither property is guaranteed when priority is given to one type of flow with respect to another. This suggests current proposals for a differentiated services Internet may lead to suboptimal utilization of network resources.

In "a queueing analysis of max-min fairness, proportional fairness and balanced fairness" by L. Massoulié, 2006, it is proposed that the performance of three usual allocations, namely max-min fairness, proportional fairness and balanced fairness, in a communication network whose resources are shared by a random number of data flows. The model consists of a network of processor sharing queues. The vector of service rates, which is constrained by some compact, convex capacity set representing the network resources, is a function of the number of customers in each queue. This function determines the way network resources are allocated. We show that this model is representative of a rich class of wired and wireless networks. We give in this general framework the stability condition of max-min fairness, proportional fairness and balanced fairness and compare their performance on a number of toy networks. Keywords: Resource allocation, flow-level modeling, stability, insensitivity.

In "randomized load balancing with general service time distributions" by J. Virtamo, 2005, it is proposed that the modularized program for treating randomized load balancing problems with general service time distributions and service disciplines. The program relies on an ansatz which asserts that any finite set of queues in a randomized load balancing scheme becomes independent as $n \rightarrow \infty$. This allows one to derive queue size distributions and other performance measures of interest. We establish the ansatz when the service discipline is FIFO and the service time distribution has a decreasing hazard rate (this includes heavy-tailed service times). Assuming the ansatz, we also obtain the following results: (i) as $n \rightarrow \infty$, the process of job arrivals at any fixed queue tends to a Poisson process whose rate depends on the size of the queue, (ii) when the service discipline at each server is processor sharing or LIFO with preemptive resume, the distribution of the number of jobs is insensitive to the service distribution, and (iii) the tail behavior of the queue-size distribution

in terms of the service distribution for the FIFO service discipline.

In "centralized content delivery infrastructure exploiting resource pools: performance models and asymptotics" by N.J. Yuan, 2007, Centralized content delivery infrastructure where a large number of storage-intensive files are replicated across several collocated servers. To achieve scalable delays in file downloads under stochastic loads, we allow multiple servers to work together as a pooled resource to meet individual download requests. In such systems basic questions include: How and where to replicate files and how significant are the gains of resource pooling over policies which use single server per request and what are the tradeoff among conflicting metrics such as delays, reliability and recovery costs, and power. How robust is performance to heterogeneity and choice of fairness criterion. Provide a simple performance model for large systems towards addressing these basic questions.

III. SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

In existing system there is scalable mean delays in file downloads under stochastic loads. Hache theorem is used for managing the files in server. Multiple server is used for storing large amount of data. Clustering technique is not used so it takes more downloading time for large files.

DISADVANTAGE:

- Pattern matching is not used
- Time delay.
- High cost due to usage of multiple servers.
- Overload of data

3.2 PROPOSED SYSTEM

In our proposed system local pattern analysis is used in different data sources together to carry out centralized computing using R Tree. Building prediction models from Big Data streams such models can adaptively adjust to the dynamic changing of the data, as well as accurately predict the pattern of the file which is searched by the users. Here in our proposed the admin uploads the files with different file formats that is audio, video, word. Keyword searched by the user is analyzed in the database and the file related to the searched query is matched with the pattern and displayed to the user. The file is portioned into number of clusters, so it reduces the downloading time.

DISADVANTAGES

- Reduce Big Data volumes to a manageable size for processing.

- Pattern matching and clustering technique is used.
- No need to use multiple servers.
- Data is managed efficiently using apriori theorem.

IV. SYSTEM IMPLEMENTATION

MODULES

1. User registration and login
2. Data upload by admin
3. Data download by user

A. USER REGISTRATION AND LOGIN

User Registration module allows a new user have to register themselves by entering their details such as Name, email id etc and access content uploaded by the admin. Already existing user sign in using their registered username and password.

B. DATA UPLOAD BY ADMIN

In this phase admin can upload data such as image, audio, video files. These files are processed by apriori theorem in database. Large amount of data can be processed easily using theorem. These files will be accessed by the users.

C. DATA DOWNLOAD BY USER

A registered user can login and download files uploaded by the admin. When the user search for a file the keyword is matched with the database and appropriate file will be downloading as clusters. The files are clustered using k-means clustering algorithm.

V. ALGORITHM EXPLANATION

(1) APRIORI ALGORITHM

The Apriori algorithm learns correlations and relations among variables in a database and is applied to a database containing a large number of transactions. Apriori is well understood, easy to implement and has many derivatives. The algorithm can be quite memory, space and time intensive when generating item sets.

ALGORITHM STEPS

Apriori (T, minSupport)

```
{  
L1= {items set};  
for (k= 2; Lk-1 !=0; k++) {  
Ck= items generated from Lk-1  
for each transaction t in database do{  
#increment the count of all generated item set in Ck that  
are contained in t(database)  
Lk = items in Ck with minSupport
```

```
}  
}  
return Uk Lk ;  
}
```

(2) K-MEANS CLUSTERING ALGORITHM

k-means clustering algorithm is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori. The main idea is to define k centers, one for each cluster. The next step is to take each point belonging to a given data set and associate it to the nearest center. It is Fast, robust and easier to understand. It gives best result when data set are distinct or well separated from each other.

VI. TECHNOLOGIES USED

6.1 JAVA

It is a Platform Independent. Java is an object-oriented programming language developed initially by James Gosling and colleagues at Sun Microsystems. The language, initially called Oak (named after the oak trees outside Gosling's office), was intended to replace C++, although the feature set better resembles that of Objective C.

WORKING OF JAVA

For those who are new to object-oriented programming, the concept of a class will be new to you. Simplistically, a class is the definition for a segment of code that can contain both data (called attributes) and functions (called methods).

When the interpreter executes a class, it looks for a particular method by the name of main, which will sound familiar to C programmers. The main method is passed as a parameter an array of strings (similar to the argv[] of C), and is declared as a static method.

To output text from the program, we execute the println method of System. Out, which is java's output stream. UNIX users will appreciate the theory behind such a stream, as it is actually standard output. For those who are instead used to the Wintel platform, it will write the string passed to it to the user's program.

Java consists of two things

- Programming language.
- Platform.

6.2 THE JAVA PROGRAMMING LANGUAGE

Java is a high-level programming language that is all of the following:

- Simple
- Object-oriented
- Distributed
- Interpreted
- Robust
- Secure
- Architecture-neutral
- Portable
- High-performance
- Multithreaded
- Dynamic

The code can bring about changes whenever felt necessary. Some of the standard needed to achieve the above-mentioned objectives are as follows:

Java is unusual in that each Java program is both compiled and interpreted. With a compiler, you translate a Java program into an intermediate language called Java byte codes – the platform independent codes interpreted by the Java interpreter. With an interpreter, each Java byte code instruction is parsed and run on the computer. Compilation happens just once; interpretation occurs each time the program is executed.

6.3 THE JAVA PLATFORM

A platform is the hardware or software environment in which a program runs. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other, hardware-based platforms. Most other platforms are described as a combination of hardware and operating system.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries (**packages**) of related components. The following figure depicts a Java program, such as an application or applet, that's running on the Java platform. As the figure shows, the Java API and Virtual Machine insulates the Java program from hardware dependencies.

As a platform-independent environment, Java can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte compilers can bring Java's performance close to that of native code without threatening portability.

(1) The Life Cycle of a JSP Page

A JSP page services requests as a servlet. Thus, the life cycle and many of the capabilities of JSP pages (in particular the dynamic aspects) are determined by Java

Servlet technology and much of the discussion in this chapter refers to functions .

When a request is mapped to a JSP page, it is handled by a special servlet that first checks whether the JSP page's servlet is older than the JSP page. If it is, it translates the JSP page into a servlet class and compiles the class. During development, one of the advantages of JSP pages over servlets is that the build process is performed automatically.

You can compile your Java program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the JVM. For example, that same Java program can be run on Windows NT, Solaris and Macintosh.

(2) Translation and Compilation

During the translation phase, each type of data in a JSP page is treated differently. Template data is transformed into code that will emit the data into the stream that returns data to the client. JSP elements are treated as follows:

- Directives are used to control how the Web container translates and executes the JSP page.
- Scripting elements are inserted into the JSP page's servlet class. See JSP Scripting Elements for details.

Both the translation and compilation phases can yield errors that are only observed when the page is requested for the first time. If an error occurs while the page is being translated (for example, if the translator encounters a malformed JSP element), the server will return a Parse Exception, and the servlet class source file will be empty or incomplete. The last incomplete line will give a pointer to the incorrect JSP element. If an error occurs while the JSP page is being compiled (for example, there is a syntax error in a script let), the server will return a Jasper Exception and a message that includes the name of the JSP page's servlet and the line where the error occurred. Once the page has been translated and compiled, the JSP page's servlet for the most part follows the servlet life cycle described in the section Servlet Life Cycle.

6.4 MySQL

MySQL is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database

structures, back up data, inspect status, and work with data records.

MySQL ships with many command line tools, from which the main interface is 'mysql' client. Third-parties have also developed tools to manage, optimize, monitor and backup a MySQL server, some listed below. All these tools work on *NIX type operating systems, and some of them also on Windows.

Like other SQL databases, MySQL does not currently comply with the full SQL standard for some of the implemented functionality, including foreign key references when using some storage engines other than the 'standard' InnoDB.

Triggers are currently limited to one per action / timing, i.e. maximum one after insert and one before insert on the same table. There are no triggers on views.

MySQL, like most other transactional relational databases, is strongly limited by hard disk performance. This is especially true in terms of write latency. Given the recent appearance of very affordable consumer grade SATA interface Solid-state drives that offer zero mechanical latency, a fivefold speedup over even an eight drive RAID array can be had for a smaller investment.

(1) NetBeans

NetBeans is an integrated development environment (IDE) for developing primarily with Java, but also with other languages, in particular PHP, C/C++, and HTML5. It is also an application platform framework for Java desktop applications and others. The NetBeans IDE is written in Java and can run on Windows, OS X, Linux, Solaris and other platforms supporting a compatible JVM.

The NetBeans Platform allows applications to be developed from a set of modular software components called modules. Applications based on the NetBeans Platform (including the NetBeans IDE itself) can be extended by third party developers.

(2) NetBeans Platform

Framework for simplifying the development of Java Swing desktop applications. The NetBeans IDE bundle for Java SE contains what is needed to start developing NetBeans plug-in and NetBeans Platform based applications; no additional SDK is required. Applications can install modules dynamically. Any application can include the Update Center module to allow users of the application to download digitally signed upgrades and new features directly into the running application. Reinstalling an upgrade or a new

release does not force users to download the entire application again.

The platform offers reusable services common to desktop applications, allowing developers to focus on the logic specific to their application.

Among the features of the platform are:

- User interface management (e.g. menus and toolbars)
- User settings management
- Storage management (saving and loading any kind of data)
- Window management
- Wizard framework (supports step-by-step dialogs)
- NetBeans Visual Library
- Integrated development tools

NetBeans IDE is a free, open-source, cross-platform IDE with built-in support for Java Programming Language.

NetBeans IDE is an open-source integrated development environment. NetBeans IDE supports development of all Java application out of the box.

Among other features are an Ant-based project system, Maven support, refactorings, version control (supporting CVS, Subversion, Git, Mercurial and Clearcase)

Modularity: All the functions of the IDE are provided by modules. Each module provides a well defined function, such as support for the Java language, editing, or support for the CVS versioning system, and SVN. NetBeans contains all the modules needed for Java development in a single download, allowing the user to start working immediately.

Modules also allow NetBeans to be extended. New features, such as support for other programming languages, can be added by installing additional modules. For instance, Sun Studio, Sun Java Studio Enterprise, and Sun Java Studio Creator from Sun Microsystems are all based on the NetBeans IDE.

VII. IMPLEMENTATION

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities. When the data is entered it will check for its validity.

Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow the dialog to guide the operating personnel in providing input.

7.1 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things

7.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making

1. Designing computer output should proceed in an organized, well thought out manner, the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives

- Convey information about past activities, current status or projections of the Future
- Signal important events, opportunities, problems, or warnings, trigger an action and confirm an action.

VIII. CONCLUSION

Our proposed concept gives better data utility, hence the file is downloading as clusters to simplify the data downloading process. Large amount of data is managed using apriori theorem and data is clustered using k-means technique by this we reduce the file downloading time.

This project summarizes existing classifying three major categories and giving real examples in each sub-category of methods also to explores the relationship and differences between different methods, helping people find proper data fusion methods to solve big data problems. Some public multi-modality datasets have been shared to facilitate further research into data fusion problems.

REFERENCES

- [1] T. Bonald, L. Massoulié, A. Proutiere, and J. Virtamo, "A queueing analysis of max-min fairness, proportional fairness and balanced fairness. Queueing systems: Theory and applications," *Queueing Syst.*, vol. 53, pp. 65–84, 2006.
- [2] T. Bonald and J. Virtamo, "Calculating the flow level performance of balanced fairness in tree networks," *Perform. Eval.*, vol. 58, no. 1, pp. 1–14, Oct. 2004.
- [3] M. Bramson, Y. Lu, and B. Prabhakar, "Randomized load balancing with general service time distributions," in *Proc. ACM SIGMETRICS*, 2010, pp. 275–286.
- [4] A. Cidon et al., "Copysets: Reducing the frequency of data loss in cloud storage," in *Proc. USENIX Annu. Tech. Conf.*, 2013, pp. 37–48.
- [5] B. Frank et al., "Collaboration opportunities for content delivery and network infrastructures," in *Recent Advances in Networking*, H. Haddadi and O. Bonaventure, Eds. New York, NY, USA: ACM, 2013, pp. 305–377.
- [6] C. Xu, T. Dacheng, and X. Chao, "A survey on multi-view learning," *arXiv preprint arXiv: 1304.5634* (2013).
- [7] P. Yang and W. Gao, "Multi-view discriminant transfer learning," *Proc. of International Joint Conference on Artificial Intelligence*, pp. 1848–1854, 2013.
- [8] N. J. Yuan, F. Zhang, D. Lian, K. Zheng, S. Yu, and X. Xie, "We know how you live: exploring the spectrum of urban lifestyles," *Proc. the first ACM Conference on Online Social Networks*, pp. 3-14, 2013.
- [9] J. Yuan, Y. Zheng, X. Xie, "Discovering regions of different functions in a city using human mobility and POIs," *Proc. ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD'12)*, pp. 186-194, 2012.
- [10] N. J. Yuan, Y. Zheng, and X. Xie, "Segmentation of Urban Areas Using Road Networks," *Technical Report MSR-TR-2012-65*, 2012.
- [11] D. Zhang, J. He, Y. Liu, L. Si, and R. Lawrence, "Multi-view transfer learning with a large margin approach. Proc. of the 17th SIGKDD conference on Knowledge Discovery and Data Mining, pp. 1208–1216, 2011.
- [12] F. Zhang, N. J. Yuan, D. Wilkie, Y. Zheng, X. Xie, "Sensing the Pulse of Urban Refueling Behavior: A Perspective from Taxi Mobility", *ACM Trans. Intelligent Systems and Technology*, submitted for publication.
- [13] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, Q. Yang, "Collaborative Filtering Meets Mobile Recommendation: A User-centered Approach," *Proc. AAAI Conf. Artificial Intelligence (AAAI'10)*, pp. 236-241, 2010.