# CHAPTER 1

# INTRODUCTION

## 1.1. Importance of Recommendation

Recommender Systems have emerged as powerful tools for helping users find and evaluate items of interest. These systems use a variety of techniques to help users identify the items that best fit their tastes or needs. While popular CF-based algorithms continue to produce meaningful, personalized results in a variety of domains, data mining techniques are increasingly being used in both hybrid systems, to improve recommendations in previously successful applications, and in stand-alone recommenders, to produce accurate recommendations in previously challenging domains. The use of data mining algorithms has also changed the types of recommendations as applications move from recommending what to consume to also recommending when to consume. While recommender systems may have started as largely a passing novelty, they clearly appear to have moved into a real and powerful tool in a variety of applications, and that data mining algorithms can be and will continue to be an important part of the recommendation process.

## 1.2. Location recommendation

Prior research has mainly investigated how to leverage spatial patterns, temporal effects spatio-temporal influence, social influence, text-based analysis, and implicit characteristics of human mobility to recommend locations. However, some of these methods require each user to have sufficient training data while others assume locations have accumulated ample textual information (e.g., tips), making it challenging to use them to tackle the cold-start problem, specifically, and recommending locations for new users. Fortunately, users are often linked to social networks, such as Twitter and Weibo, which probably collect rich semantic content from users. This semantic content is likely to imply user interest, an essential element for capturing users’ visiting behavior.

Therefore, they can be exploited to address the cold-start challenge and even improve location recommendation. A typical method is to feed them into traditional explicit-feedback content-aware recommendation frameworks, such as LibFM, SVD Feature, regression-based latent factor model or Match Box. These frameworks require drawing negative samples from unvisited locations for better learning performance, since a user’s negative preference for locations is not observable in human mobility data. However, it has been empirically shown that sampling-based frameworks do not perform as well as an algorithm that treats all unvisited locations as negative yet assigns them a lower preference confidence, since the latter one deals with the sparsity issues better.

## 1.3. Challenges and issue in recommendation system

### 1.3.1. Cold Start Problem

The term derives from cars. When it’s really cold, the engine has problems with starting up, but once it reaches its optimal operating temperature, it will run smoothly. With recommendation engines, the “cold start” simply means that the circumstances are not yet optimal for the engine to provide the best possible results. In ecommerce, there are two distinct categories of cold start: product cold start and user cold starts. News sites, auction sites, ecommerce stores and classified sites all experience the product cold start. The user or visitor cold start simply means that a recommendation engine meets a new visitor for the first time. Because there is no user history about the user, the system doesn’t know the personal preferences of the user. Getting to know your visitors is crucial in creating a great user experience for them.

### 1.3.2. Data Sparsity

In practice, many commercial recommender systems are based on large datasets. As a result, the user-item matrix used for collaborative filtering could be extremely large and sparse, which brings about the challenges in the performances of the recommendation. One typical problem caused by the data sparsity is the cold start problem. As collaborative filtering methods recommend items based on users' past preferences, new users will need to rate sufficient number of items to enable the system to capture their preferences accurately and thus provides reliable recommendations.

### 1.3.3. Scalability

As the numbers of users and items grow, traditional CF algorithm will suffer serious scalability problems. For example, with tens of millions of customers and millions of items, a CF algorithm with the complexity of n is already too large. As well, many systems need to react immediately to online requirements and make recommendations for all users regardless of their purchases and ratings history, which demands a higher scalability of a CF system. Large web companies such as Twitter use clusters of machines to scale recommendations for their millions of users, with most computations happening in very large memory machines.

### 1.3.4. Gray Sheep

Gray sheep refers to the users whose opinions do not consistently agree or disagree with any group of people and thus do not benefit from collaborative filtering. Black sheep are the opposite group whose idiosyncratic tastes make recommendations nearly impossible. Although this is a failure of the recommender system, non-electronic recommenders also have great problems in these cases, so black sheep is an acceptable failure.

In addition, based on this evaluation, we find that user profiles and semantic content can make significant improvements over the counterpart without taking them into account. In addition to the warm-start evaluation, we also perform a cold-start evaluation with a user-based 5-fold cross validation by splitting users into five non-overlapping groups. The results indicate that both user profiles and semantic content are useful for tackling the cold-start problem in location recommendation based on human mobility data, and that user profiles are more effective than semantic content.

By understanding the need of recommendation system in current generation and its issue were also discussed. Hence a method to overcome this issue has been proposed through machine learning method called NLP. Through this NLP could recommend best place to new user by overcoming issues in recommendation process. Efficient classification SVM is used for better result classification.

# CHAPTER 2

# LITERATURE REVIEW

**TITLE:** Content-aware collaborative filtering for location recommendation based on human mobility data [1]

**AUTHORS:** Defu Lian, Yong Ge, Fuzheng Zhang

Location recommendation plays an essential role in helping people find places they are likely to enjoy. Though some recent research has studied how to recommend locations with the presence of social network and geographical information, few of them addressed the cold-start problem, specifically, recommending locations for new users. Because the visits to locations are often shared on social networks, rich semantics (e.g., tweets) that reveal a person's interests can be leveraged to tackle this challenge. A typical way is to feed them into traditional explicit-feedback content-aware recommendation methods (e.g., LibFM). However, prior studies have empirically shown that sampling-based methods don't perform as well as a method that considers all unvisited locations as negative but assigns them a lower confidence. To this end, we propose an Implicit-feedback based Content-aware Collaborative Filtering (ICCF) framework to incorporate semantic content and steer clear of negative sampling. For efficient parameter learning, we develop a scalable optimization algorithm, scaling linearly with the data size and the feature size. Furthermore, we offer a good explanation to ICCF, such that the semantic content is actually used to refine user similarity based on mobility. Finally, we evaluate ICCF with a large-scale LBSN dataset where users have profiles and text content. The results show that ICCF outperforms LibFM of the best configuration, and that user profiles and text content are not only effective at improving recommendation but also helpful for coping with the cold-start problem.

**TITLE:** Collaborative filtering meets mobile recommendation: A user-centered approach [2]

**AUTHORS:** Vincent W. Zheng, Bin Cao, Yu Zheng, Xing Xie, Qiang Yang

With the increasing popularity of location tracking services such as GPS, more and more mobile data are being accumulated. Based on such data, a potentially useful service is to make timely and targeted recommendations for users on places where they might be interested to go and activities that they are likely to conduct. For example, a user arriving in Beijing might wonder where to visit and what she can do around the Forbidden City. A key challenge for such recommendation problems is that the data we have on each individual user might be very limited, while to make useful and accurate recommendations, we need extensive annotated location and activity information from user trace data. In this paper, we present a new approach, known as user-centered collaborative location and activity filtering (UCLAF), to pull many users' data together and apply collaborative filtering to find like-minded users and like-patterned activities at different locations. We model the user location-activity relations with a tensor representation, and propose a regularized tensor and matrix decomposition solution which can better address the sparse data problem in mobile information retrieval.

**TITLE:** Exploiting geographical influence for collaborative point-of-interest recommendation [3]

**AUTHORS:** Mao Ye, Peifeng Yin, Wang-Chien Lee

In this paper, we aim to provide a point-of-interests (POI) recommendation service for the rapid growing location-based social networks (LBSNs), e.g., Foursquare, Whrrl, etc. Our idea is to explore user preference, social influence and geographical influence for POI recommendations. In addition to deriving user preference based on user-based collaborative filtering and exploring social influence from friends, we put a special emphasis on geographical influence due to the spatial clustering phenomenon exhibited in user check-in activities of LBSNs. We argue that the geographical influence among POIs plays an important role in user check-in behaviors and model it by power law distribution. Accordingly, we develop a collaborative recommendation algorithm based on geographical influence based on naive Bayesian. Furthermore, we propose a unified POI recommendation framework, which fuses user preference to a POI with social influence and geographical influence. Finally, we conduct a comprehensive performance evaluation over two large-scale datasets.

**TITLE:** Modeling user mobility for location promotion in location-based social networks [4]

**AUTHORS:** Wen-Yuan Zhu, Wen-Chih Peng and Ling-Jyh Chen

With the explosion of smartphones and social network services, location-based social networks (LBSNs) are increasingly seen as tools for businesses (e.g., restaurants, hotels) to promote their products and services. In this paper, we investigate the key techniques that can help businesses promote their locations by advertising wisely through the underlying LBSNs. In order to maximize the benefit of location promotion, we formalize it as an influence maximization problem in an LBSN, i.e., given a target location and an LBSN, which a set of *k* users (called seeds) should be advertised initially such that they can successfully propagate and attract most other users to visit the target location. Existing studies have proposed different ways to calculate the information propagation probability, that is how likely a user may influence another, in the settings of static social network. However, it is more challenging to derive the propagation probability in an LBSN since it is heavily affected by the target location and the user mobility, both of which are dynamic and query dependent. This paper proposes two user mobility models, namely Gaussian-based and distance-based mobility models, to capture the check-in behavior of individual LBSN user, based on which location-aware propagation probabilities can be derived respectively. Extensive experiments based on two real LBSN datasets have demonstrated the superior effectiveness of our proposals than existing static models of propagation probabilities to truly reflect the information propagation in LBSNs.

**TITLE:** Learning Geographical Preferences for Point-of-Interest Recommendation [5]

**AUTHORS:** Bin Liu, Yanjie Fu, Zijun Yao, Hui Xiong

The problem of point of interest (POI) recommendation is to provide personalized recommendations of places of interests, such as restaurants, for mobile users. Due to its complexity and its connection to location based social networks (LBSNs), the decision process of a user choose a POI is complex and can be influenced by various factors, such as user preferences, geographical influences, and user mobility behaviors. While there are some studies on POI recommendations, it lacks of integrated analysis of the joint effect of multiple factors. To this end, in this paper, we propose a novel geographical probabilistic factor analysis framework which strategically takes various factors into consideration. Specifically, this framework allows to capture the geographical influences on a user’s check-in behavior. Also, the user mobility behaviors can be effectively exploited in the recommendation model. Moreover, the recommendation model can effectively make use of user check-in count data as implicitly user feedback for modeling user preferences. The experimental results on real-world LBSNs data show that the proposed recommendation method outperforms state-of-the-art latent factor models with a significant margin.

**TITLE:** Exploring temporal effects for location recommendation on location-based social networks [6]

**AUTHORS:** Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu

Location-based social networks (LBSNs) have attracted an inordinate number of users and greatly enriched the urban experience in recent years. The availability of spatial, temporal and social information in online LBSNs offers an unprecedented opportunity to study various aspects of human behavior, and enable a variety of location-based services such as location recommendation. Previous work studied spatial and social influences on location recommendation in LBSNs. Due to the strong correlations between a user's check-in time and the corresponding check-in location, recommender systems designed for location recommendation inevitably need to consider temporal effects. In this paper, we introduce a novel location recommendation framework, based on the temporal properties of user movement observed from a real-world LBSN dataset. The experimental results exhibit the significance of temporal patterns in explaining user behavior, and demonstrate their power to improve location recommendation performance.

**TITLE:** Time aware point-of-interest recommendation [7]

**AUTHORS:** Quan Yuan, Gao Cong, Zongyang Ma

The availability of user check-in data in large volume from the rapid growing location-based social networks (LBSNs) enables many important location-aware services to users. Point-of-interest (POI) recommendation is one of such services, which is to recommend places where users have not visited before. Several techniques have been recently proposed for the recommendation service. However, no existing work has considered the temporal information for

POI recommendations in LBSNs. We believe that time plays an important role in POI recommendations because most users tend to visit different places at different time in a day, e.g., visiting a restaurant at noon and visiting a bar at night. In this paper, we define a new problem, namely, the time-aware POI recommendation, to recommend POIs for a given user at a specified time in a day. To solve the problem, we develop a collaborative recommendation model that is able to incorporate temporal information. Moreover, based on the observation that users tend to visit nearby POIs, we further enhance the recommendation model by considering geographical information. Our experimental results on two real-world datasets show that the proposed approach outperforms the state-of-the-art POI recommendation methods substantially.

**TITLE:** Graph-based point-of-interest recommendation with geographical and temporal influences [8]

**AUTHORS:** Quan Yuan, Gao Cong and Aixin Sun

The availability of user check-in data in large volume from the rapid growing location-based social networks (LBSNs) enables a number of important location-aware services. Point-of-interest (POI) recommendation is one of such services, which is to recommend POIs that users have not visited before. It has been observed that: (i) users tend to visit nearby places, and (ii) users tend to visit different places in different time slots, and in the same time slot, users tend to periodically visit the same places. For example, users usually visit a restaurant during lunch hours, and visit a pub at night. In this paper, we focus on the problem of time-aware POI recommendation, which aims at recommending a list of POIs for a user to visit at a given time. To exploit both geographical and temporal influences in time-aware POI recommendation, we propose the Geographical-Temporal influences Aware Graph (GTAG) to model check-in records, geographical influence and temporal influence. For effective and efficient recommendation based on GTAG, we develop a preference propagation algorithm named Breadth-first Preference Propagation (BPP). The algorithm follows a relaxed breath-first search strategy, and returns recommendation results within at most 6 propagation steps. Our experimental results on two real-world datasets show that the proposed graph-based approach outperforms state-of-the-art POI recommendation methods substantially.

**TITLE:** A random walk around the city: New venue recommendation in location-based social networks [9]

**AUTHORS:** Anastasios Noulas, Salvatore Scellato, Neal Lathia, Cecilia Mascolo

The popularity of location-based social networks available on mobile devices means that large, rich datasets that contain a mixture of behavioral (users visiting venues), social (links between users), and spatial (distances between venues) information are available for mobile location recommendation systems. However, these datasets greatly differ from those used in other online recommender systems, where users explicitly rate items: it remains unclear as to how they capture user preferences as well as how they can be leveraged for accurate recommendation. This paper seeks to bridge this gap with a three-fold contribution. First, we examine how venue discovery behavior characterizes the large check-in datasets from two different location-based social services, Foursquare and Go Walla: by using large-scale datasets containing both user check-ins and social ties, our analysis reveals that, across 11 cities, between 60% and 80% of users' visits are in venues that were not visited in the previous 30 days. We then show that, by making constraining assumptions about user mobility, state-of-the-art filtering algorithms, including latent space models, do not produce high quality recommendations. Finally, we propose a new model based on personalized random walks over a user-place graph that, by seamlessly combining social network and venue visit frequency data, obtains between 5 and 18% improvement over other models. Our results pave the way to a new approach for place recommendation in location-based social systems.

**TITLE:** Content-based recommender systems: State of the art and trends [10]

**AUTHORS:** Pasquale Lops

Recommender systems have the effect of guiding users in a personalized way to interesting objects in a large space of possible options. *Content-based* recommendation systems try to recommend items similar to those a given user has liked in the past. This chapter provides an overview of content-based recommender systems, with the aim of imposing a degree of order on the diversity of the different aspects involved in their design and implementation. The first part of the chapter presents the basic concepts and terminology of content-based recommender systems, a high-level architecture, and their main advantages and drawbacks. The second part of the chapter provides a review of the state of the art of systems adopted in several application domains, by thoroughly describing both classical and advanced techniques for representing items and user profiles. The last part of the chapter discusses trends and future research which might lead towards the next generation of systems, by describing the role of User Generated Content as a way for taking into account evolving vocabularies, and the challenge of feeding users with serendipitous recommendations, that is to say surprisingly interesting items that they might not have otherwise discovered.

# CHAPTER 3

# SYSTEM ANALYSIS

## 3.1. EXISTING SYSTEM

Location recommendation has recently become a popular research topic, with the support of massive data. Prior research has mainly investigated how to leverage spatial patterns, temporal effects, spatio-temporal influence, social influence, text-based analysis, and implicit characteristics of human mobility to recommend locations. A novel scalable Implicit-feedback based Content-aware Collaborative Filtering (ICCF) framework. It steers clear of sampling negative locations, by treating all unvisited locations as negative and proposing a sparse and rank-one weighting configuration for modeling preference confidence. This sparse and rank one weighting configuration not only assigns vastly varying confidence to visited and unvisited locations, but also subsumes three previously developed different weighting schemes for unvisited locations and naturally introduces a novel mixed weighting scheme. ICCF not only improves location recommendation, but also addresses the cold-start problems of both new users and new locations.

### 3.1.1. LIMITATIONS

- A user's negative preference for locations is not observable in human mobility data.
- Accuracy is low
- Low performance and efficiency

## 3.2. PROPOSED SYSTEM

Recommending result according to user query is a difficult task. Recommendation process takes place through feedback, review of user in different format like rating, like or dislike and commands. Analyzing user reviews is one of effective methods to generate efficient recommendation to new user. Therefore, it includes content-based analysis which attains maximum accuracy in result. Hence in our proposed user reviews are taken into consideration and loaded into database for further processing. Initially preprocessing is implemented to avoid unwanted noises and undergone for next meaning analysis. This process has been done through NLP which identify whether user review is positive or negative command. Include in our database if processed command is positive and suggest it for further recommendation process for new user. In our proposed work content-based location recommendation is implemented through NLP to attain maximum accuracy in results. This semantic content is likely to imply user interest, an essential element for capturing users' visiting behavior. For accurate classification of user review as either good or bad efficient classification SVM is used.

### 3.2.1. EXPECTED MERITS

- **Efficient location recommendation** based on user review comments location recommendation has been done which generates optimal solution from a large database.
- **Avoid cold start problem** without any knowledge of user interest through our method we can suggest best valid results according to user query.
- **Content based analysis** instead of page ranking, rating in our method content-based recommendation which results in efficient recommendation process.
- **Accuracy is high** compared to other recommendation process our method attains maximum accuracy in analyzing best location suggestion to a user.

# CHAPTER 4

# SYSTEM SPECIFICATION

## 4.1. HARDWARE REQUIREMENTS (Minimum requirement)

The section of hardware configuration is an important task related to the software development insufficient random-access memory may affect adversely on the speed and efficiency of the entire system. The process should be powerful to handle the entire operations. The hard disk should have sufficient capacity to store the file and application.

- System : Pentium Dual Core
- Hard Disk : 120 GB
- Ram : 1GB

## 4.2. SOFTWARE REQUIREMENTS

A major element in building a system is the section of compatible software since the software in the market is experiencing in geometric progression. Selected software should be acceptable by the firm and one user as well as it should be feasible for the system. This document gives a detailed description of the software requirement specification. The study of requirement specification is focused specially on the functioning of the system. It allows the developer or analyst to understand the system, function to be carried out the performance level to be obtained and corresponding interfaces to be established.

- Operating system : Windows 7
- Coding Language : JAVA/J2EE
- Tool : Netbeans 8.2
- Database : MYSQL

## 4.3. SOFTWARE DESCRIPTION

### a) INTRODUCTION TO FRONT END

The software requirement specification is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description as functional representation, a representation of system behavior, an indication of performance requirements and design constraints, appropriate validation criteria.

### b) JAVA

Java was conceived by James Gosling, Patrick Naughton, Chris Wrath, Ed Frank, and Mike Sheridan at Sun Micro system. It is a platform independent programming language that extends its features wide over the network. It's a light weight package, as they are not implemented by platform-specific code.

Java is also unusual in that each Java program is both compiled and interpreted. With a compiler, you translate a Java program into an intermediate

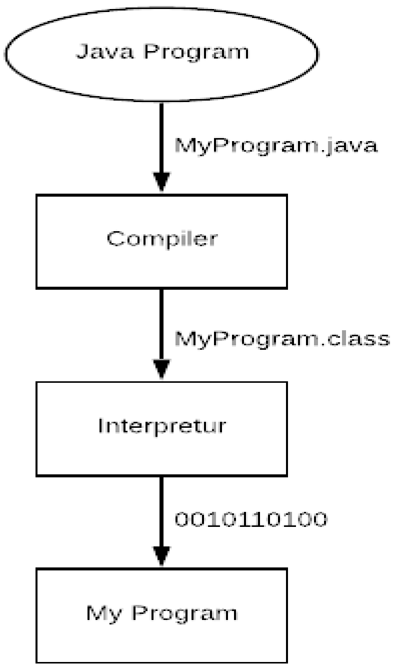



Fig. 4.1 Java Program Execution

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. This figure illustrates how this works.

Java byte codes can be considered as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

**c) JSP**

JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to Servlet because it provides more functionality than servlet such as expression language, JSTL, etc.

A JSP page consists of HTML tags and JSP tags. The JSP pages are easier to maintain than Servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tags, etc.

**d) INTRODUCTION TO BACK END**

**MICROSOFT SQL SERVER 2005**

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

**MYSQL DATABASE**

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. MySQL is becoming so popular because of many good reasons

- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

# CHAPTER 5

# MODULE DESCRIPTION

Implementation is the stage in the project where the theoretical design is turned into a working system and is giving confidence on the new system for the users, which it will work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, an evaluation, of change over methods. The more complex system being implemented, the more involved will be the system analysis and the design effort required just for implementation.

An implementation co-ordination committee based on policies of individual organization has been appointed. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. Implementation is the final and important phase, the most critical stage in achieving a successful new system and in giving the users confidence. That the new system will work be effective. The system can be implemented only after through testing is done and if it found to working according to the specification. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle certain type of transactions while using the new system.

## MODULES

- Content feedback from user
- SVM
- Identifying positive feedback
- User query and result

## 5.1. Content feedback from user

In this module, the feedbacks collected from users through social networks by posting images and giving comments with respect to location information. Therefore, data are gathered from various users are considered and loaded in our database through our application will used for analysis. The embracement of the web into our daily life activities in this contemporary period has become almost inevitable and quite numbers of populace rely on the web for different purpose range from placing on their view and read others view while also commenting on such views, e-learning, e-banking, e-library and e-commerce etc. The number of available documents on the web is enough to improve the diverse ways of educating and research need of the public now and then. In this system, data is collected on the basis of our visited web pages, our activities in social networks, smart phones and through the many sensors of the physical world. It is this stream that forms the basis of Big Data. A data stream (or flow of different streams) basically, without interpretation, has less value, but based on analysis creates information that we can use, so the data becomes valuable. In our work it will focus on making use of context-based approach in addition to CF approach to recommend quality content to its users. It would be exploiting available contextual information, analyzing and summarizing user queries, and linking the metadata like tags and feedback to a richer information model to recommend content.

## 5.2. Support Vector Machine

Support Vector Machine (SVM) is a noteworthy methodology for characterizing high-dimensional information with the utilization of Structural Risk Minimization (SRM) rule. SVM has been passed on as a discriminative classifier which is further precise than most previous order models. SVM gains the ideal hyper plane those parts preparing information focuses from various classes by expanding the arrangement edge. Also, SVM is utilized to information focuses with nonlinear choice surfaces by connecting with a framework recognized as the part technique that plans the information to a higher dimensional component space, where a direct isolating hyper plane can be dispatch.

### 5.2.1. Natural language processing

#### Morphology

Morphology is the first stage of analysis once input has been received. It looks at the ways in which words break down into their components and how that affects their grammatical status. Morphology is mainly useful for identifying the parts of speech in a sentence and words that interact together. The following quote from Forsberg gives a little background on the field of morphology.

Morphology is a systematic description of words in a natural language. It describes a set of relations between words' surface forms and lexical forms. A word's surface form is its graphical or spoken form, and the lexical form is an analysis of the word into its lemma (also known as its dictionary form) and its grammatical description. This task is more precisely called inflectional morphology.

Being able to identify the part of speech is essential to identifying the grammatical context a word belongs to. In English, regular verbs have a ground form with a limited set of modifications, however, irregular verbs do not follow these modification rules, and greatly increase the complexity of a language. The information gathered at the morphological stage prepares the data for the syntactical stage which looks more directly at the target language's grammatical structure.

*1) Syntax*

Syntax involves applying the rules of the target language's grammar, its task is to determine the role of each word in a sentence and organize this data into a structure that is more easily manipulated for further analysis. Semantics are the examination of the meaning of words and sentences.

a) Grammar

In English, a statement consists of a noun phrase, a verb phrase, and in some cases, a prepositional phrase. A noun phrase represents a subject that can be summarized or identified by a noun. This phrase may have articles and adjectives and/or an embedded verb phrase as well as the noun itself. A verb phrase represents an action and may include an imbedded noun phrase along with the verb. A prepositional phrase describes a noun or verb in the sentence. The majority of natural languages are made up of a number of parts of speech mainly: verbs, nouns, adjectives, adverbs, conjunctions, pronouns and articles.

b) Parsing

Parsing is the process of converting a sentence into a tree that represents the sentence's syntactic structure. The statement: "The green book is sitting on the desk" consists of the noun phrase: "The green book" and the verb phrase: "is sitting on the desk." The sentence tree would start at the sentence level and break it down into the noun and verb phrase. It would then label the articles, the adjectives and the nouns. Parsing determines whether a sentence is valid in relation to the language's grammar rules.

c) Semantics

It builds up a representation of the objects and actions that a sentence is describing and includes the details provided by adjectives, adverbs and propositions. This process gathers information vital to the pragmatic analysis in order to determine which meaning was intended by the user.

d) Pragmatics

Pragmatics is “the analysis of the real meaning of an utterance in a human language, by disambiguating and contextualizing the utterance”. This is accomplished by identifying ambiguities encountered by the system and resolving them using one or more types of disambiguation techniques.

*2) Keyword based searching*

Information retrieval is the process of gathering information by using keywords from the relevant document and that document can be unstructured or structured data. It hides its complexity from user by providing abstract view. As user don't have any knowledge about schema and any other query processing language, he can search through abstract interface by putting keywords. By using Keyword Search user can submit keyword to search engines (Internet Search) or structured data and in turn it returns a list of documents to user according to ranking. Ranking of documents are provided based on the keywords match and occurrence of keyword match in particular document. Ranking is provided in descending order of occurrence of keyword match and the document with maximum occurrence get higher priority.

**5.3. Identifying positive feedback**

In this module, the loaded comment database is utilized for analysis of positive comments with respect to location. Data is extracted first from online sources like weblogs or websites. Data consist of comments and feedback posted by users on the website. The data which is fetched is unstructured and not in a useful form. This data is properly extracted from web pages and stored in database. These comments are stored according to location. The comments are stored in string format and are processed and broken into tokens to analyze each word in the string. These tokens are then sorted to remove repetitive words and prepositions which are not useful for determining polarity of comments. The positive and negative responses are grouped into separate database which were used for deciding the comments to be positive or negative. The useful tokens are stored in a list to determine their polarity i.e. positive or negative comments. The tokens are then used to determine for which feature the comment is made and the accordingly the ratings are assigned to that feature. The comment is related to a particular feature or not is determined by the words used in the comments provided. Based on user feedbacks either it is good place to visit or not is decided through average calculation. If maximum average calculation attains negative value then it will be not considered to be recommended.

### 5.4. User query and result

In this module, user enters their keyword and search in our analyzed and processed database. The keyword search plays a major role in searching particular content from a huge database. Most relevant content should be extracted from a huge database and the recommendation of particular content is done through NLP. In general reviews are available in different format such as rating, likes and comments among these three comment-based reviews is most valuable therefore processing it automatically through machine is not easy task. In our proposed we are going to process it through Machine learning. Here recommendation has been done through recommending best places suggested by users after visiting those places. Therefore, most related content has been retrieved to respective user.

## 5.5. ARCHITECTURAL DESIGN

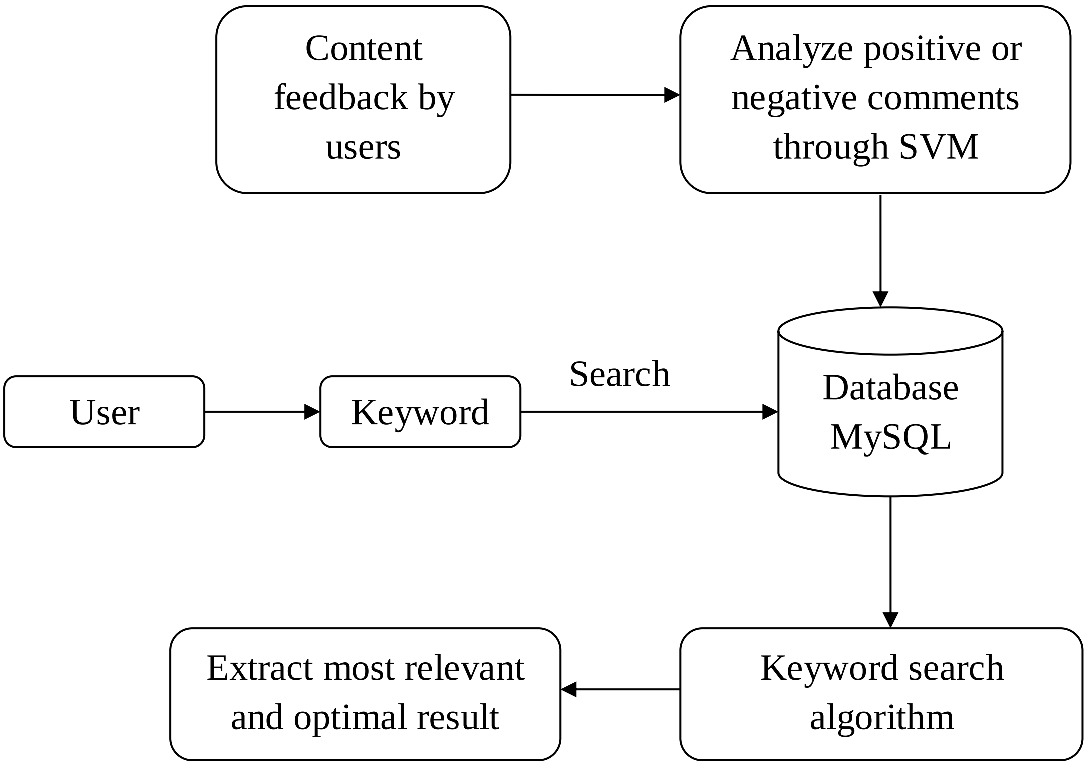



Fig. 5.1 Architectural design

# CHAPTER 6
# FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

The three key considerations involved in the feasibility analysis are

- Economical Feasibility
- Technical Feasibility
- Social Feasibility

## 6.1. ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have. The developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

## 6.2. TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

### 6.3. SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

# CHAPTER 7

# CONCLUSION

A Machine Learning based framework for content-aware collaborative filtering from user feedback commands is proposed. Our experiment results indicate that NLP is superior to five competing baselines, including two state-of-the-art location recommendation algorithms and ranking-based factorization machine. Initially user feedbacks were collected from different websites are preprocessed and stored in database for processing. In preprocessing the unwanted content, irrelevant data are removed from database. Once preprocessed content is utilized by NLP for separating the content as string and analyzing it through both syntax and semantic analyzes. The meaning of words was analyzed and classified. By studying the effects of user profiles and semantic content, we find that they improve recommendation in warm-start cases and help address the cold-start problems. Once commands nature has been identified as either positive or negative through NLP next it was subjected to classification. SVM will efficiently classifies its category accurately and used for better prediction compared to existing system. Finally, user will retrieve their data based on recommended content through keyword-based search process which retrieve data with more relevancy.

# APPENDIX 1

## SOURCE CODE

### signup.jsp

```
<%@page import="com.controller.userLogin.User_Model"%>
<html>
    <head>
        <style>
            *{
                box-sizing: border-box;
            }
            html{
                height: 100%;
            }
            body{
                padding: 60px 15px;
                min-height: 100%;
                background-image: linear-gradient(61deg, #ff512f 0%, #dd2476
100%);
                background-size: cover;
            }

            /* Form */
            .form{
                max-width: 460px;
                padding: 30px;
                margin: 0 auto;
                border-radius: 4px;
                box-shadow: 0 0 10px rgba(0,0,0,.2);
                background: #ffffff;
            }
            /* Row */
            .form__row{
                display: flex;
                width: 100%;
                justify-content: center;
            }
            .form__row:not(:first-child){
                margin-top: 15px;
            }
            /* Field */
            .form__input{
```

```
    width: 100%;
    padding: 10px 15px;
    border: 0;
    border-radius: 4px;
    background-color: #eee;
    font-size: 14px;
    line-height: 20px;
    color: #7a7b7f;
    transition: box-shadow 0.2s ease;
}
.form__input:focus{
    outline: none;
    box-shadow: 0 0 3px rgba(0,0,0,0.3);
}
.form__input ~ .form__input{
    margin-left: 15px;
}
/* Submit */
.form__submit{
    position: relative;
    width: 100%;
    height: 50px;
    border: 0;
    border-radius: 4px;
    background-color: #ea355a;
    box-shadow: 0 3px #bd1962;
    font-size: 18px;
    font-weight: bold;
    color: #fff;
    cursor: pointer;
    outline: none;
    transition: background 0.5s ease;
}
.form__submit:hover{
    background: #ff512f;
}
.form__submit:active{
    box-shadow: none;
    top: 2px;
    box-shadow: 0 1px #bd1962;
}
/* Reset */
.form__reset{
```

```
            border: 1px solid #eee;
            border-radius: 4px;
            background-color: transparent;
            font-size: 12px;
            line-height: 20px;
            color: #7a7b7f;
            outline: none;
            cursor: pointer;
            transition: border-color 0.5s ease;
        }
        .form__reset:hover{
            border-color: #7a7b7f;
        }

    </style>
  </head>
  <body>
    <%
        Boolean isSuccess = (Boolean) request.getAttribute("isSuccess");
        User_Model u = (User_Model) request.getAttribute("enteredData");
        Boolean ReturnData = false;
    %>

    <%
        if (isSuccess != null) {
            if (isSuccess) {
                ReturnData = !ReturnData;
    %>
    <% response.sendRedirect("index.jsp");
            }
        }%>
    <form action="SignUpController" method="post" class="form">
        <div class="form__row">
            <input class="form__input" value="<%if(ReturnData)
out.println(u.getStaff_first_name());%>" required="" type="text" name="first-
name" placeholder="First name">
            <input class="form__input" value="<%if(ReturnData)
out.println(u.getStaff_middle_name());%>"  type="text" name="middle-name"
placeholder="Middle name">
        </div>
        <div class="form__row">
            <input class="form__input" type="text" name="last-name"
placeholder="Last Name" required="" value="<%if(ReturnData)
```

```
out.println(u.getStaff_last_name());%>">
            </div>
            <div class="form__row">
                <input class="form__input" type="password" name="password"
placeholder="password" required="">
            </div>
            <div class="form__row">
                <button class="form__submit" type="submit">Sign up for
free</button>
            </div>
            <div class="form__row">
                <button class="form__reset" type="reset">Reset form</button>
            </div>
        </form>
    </body>
</html>
```

**login.jsp**

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Mobile Recommender login</title>
        <link href="css/login.css" rel="stylesheet" type="text/css"/>
    </head>
    <body>

        <div class="container">
<section id="content">
<form action="userCOntroller" method="POST">
<h1>Login Form</h1>
<div>
                            <input type="text" placeholder="Username" required=""
id="username" name="user"   />
</div>
<div>
                            <input type="password" name = "pass"
placeholder="Password" required="" id="password"  />
</div>
<div><input type="submit" value="Log in" />
<a href="#">Lost your password?</a>
```

```
                                <a href="signup.jsp">Register</a>
</div>
</form><!-- form -->
</section><!-- content -->
</div><!-- container -->
    </body>
```

**registercontroller.java**

```
package com.registerFone.coln;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.io.PrintWriter;
import java.nio.file.Paths;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.Statement;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import javax.servlet.ServletException;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.Part;
//@WebServlet("/uploadImage")
@MultipartConfig(maxFileSize = 16177216)
public class RegisterController extends HttpServlet {
   // MaiApp ma=new MaiApp();

    protected void processRequest(HttpServletRequest request,
HttpServletResponse response)
            throws ServletException, IOException {
    }
    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on
the + sign on the left to edit the code.">
```

```
   @Override
   protected void doGet(HttpServletRequest request, HttpServletResponse
response)
           throws ServletException, IOException {
       processRequest(request, response);
   }

   public  double maindemo(String args) throws IOException {

//String text = "Those who find ugly meanings in beautiful things are corrupt
without being charming.";
String text=args;
SentimentAnalyzer sentimentAnalyzer = new SentimentAnalyzer();
sentimentAnalyzer.initialize();
SentimentResult sentimentResult =
sentimentAnalyzer.getSentimentResult(text);
               double sm= sentimentResult.getSentimentScore();
System.out.println("Sentiment Score: " +
sentimentResult.getSentimentScore());
System.out.println("Sentiment Type: " + sentimentResult.getSentimentType());
System.out.println("Very positive: " +
sentimentResult.getSentimentClass().getVeryPositive()+"%");
System.out.println("Positive: " +
sentimentResult.getSentimentClass().getPositive()+"%");
System.out.println("Neutral: " +
sentimentResult.getSentimentClass().getNeutral()+"%");
System.out.println("Negative: " +
sentimentResult.getSentimentClass().getNegative()+"%");
System.out.println("Very negative: " +
sentimentResult.getSentimentClass().getVeryNegative()+"%");
               return sm;

}
   @Override
   protected void doPost(HttpServletRequest request, HttpServletResponse
response)
           throws ServletException, IOException {
    //   processRequest(request, response);


  //     response.setContentType("text/html;charset=UTF-8");
       PrintWriter out = response.getWriter();
       try {
```

```
String name=request.getParameter("name");
System.out.println("Name "+name);
 //String area=request.getParameter("area");
//  String name=
//    System.out.println("name of cookies "+name);
String place=request.getParameter("brand");
String description=request.getParameter("description");
//description.contains("")
double sm=maindemo(description);
double sm1=0;
String area=request.getParameter("area");

if(sm==1.0)
{
   sm1=-1.0;
}
else if(sm==2.0)
{
   sm1=0;
}
else
{
   sm1=1.0;

}
Class.forName("com.mysql.jdbc.Driver");
Connection cn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/colz_db", "root",
"root");
Part image1 = request.getPart("ImagePath");
InputStream inputStream = image1.getInputStream();
byte[] buffer = new byte[inputStream.available()];
inputStream.read(buffer);
String image2=Paths.get(image1.getSubmittedFileName()).toString();
String image=
Paths.get(image1.getSubmittedFileName()).getFileName().toString();//
image1.getSubmittedFileName();
System.out.println(request.getPart("ImagePath").toString());
System.out.println("image 1:"+image);
System.out.println("image :"+image2);
String path="C:\\demo\\"+image;
System.out.println("path :"+path);
File fsd=new File(path);
```

```
            File targetFile = new File(path);
            OutputStream outStream = new FileOutputStream(targetFile);
            outStream.write(buffer);
            image1.getSize();
             PreparedStatement ps = cn.prepareStatement("insert into location
(image,place,description,path,area,name,rating,image1) values(?,?,?,?,?,?,?,?)");
   InputStream is = image1.getInputStream();
   ps.setString(1,image);
                        ps.setString(2,place);
                        ps.setString(3,description);
                        ps.setString(4,path);
                        ps.setString(5,area);
                        ps.setString(6,name);
                        ps.setDouble(7,sm1);
                        ps.setBlob(8,is);
   int a = ps.executeUpdate();
            Statement stmt=cn.createStatement();
        image1.delete();
            if (a==1) {
                response.sendRedirect("index.jsp");
            } else {
                response.sendRedirect("registernewMobile.jsp");
            }
        }
        catch(Exception e)
        {
            System.out.println(e);
        }
        finally
            out.close();
        }
    }
    @Override
    public String getServletInfo() {
        return "Short description";
    }

}
```

# APPENDIX 2

## SCREENSHOTS

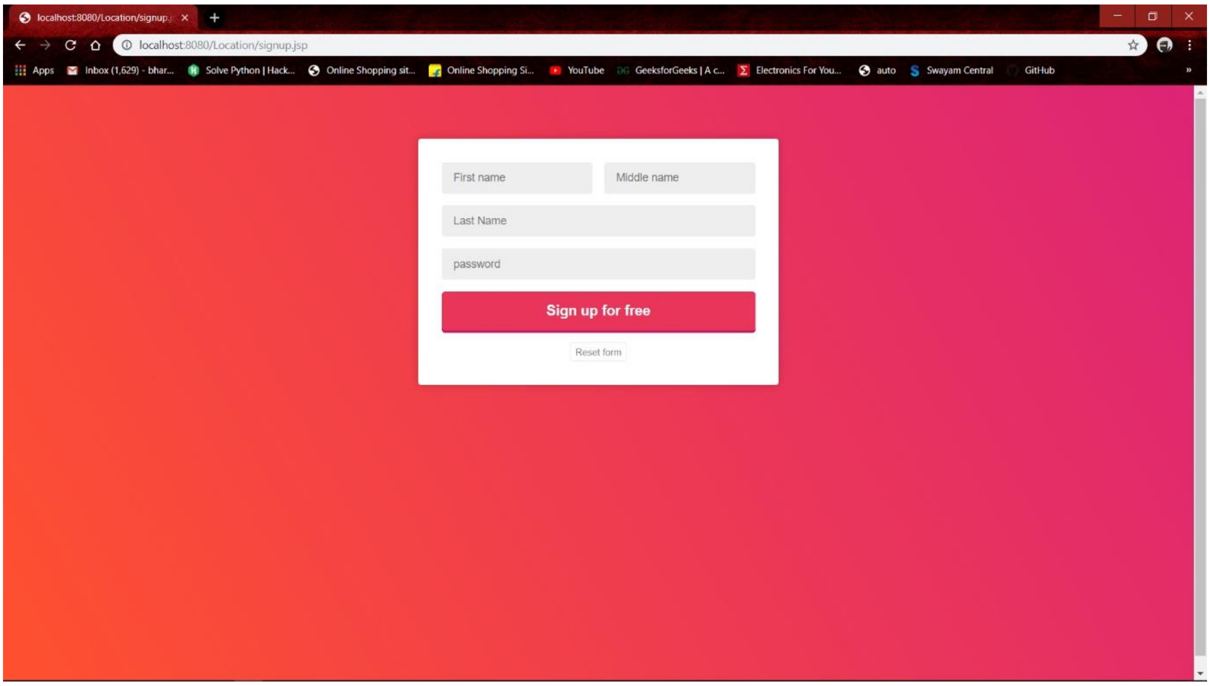



Fig.A2.1. Signup page

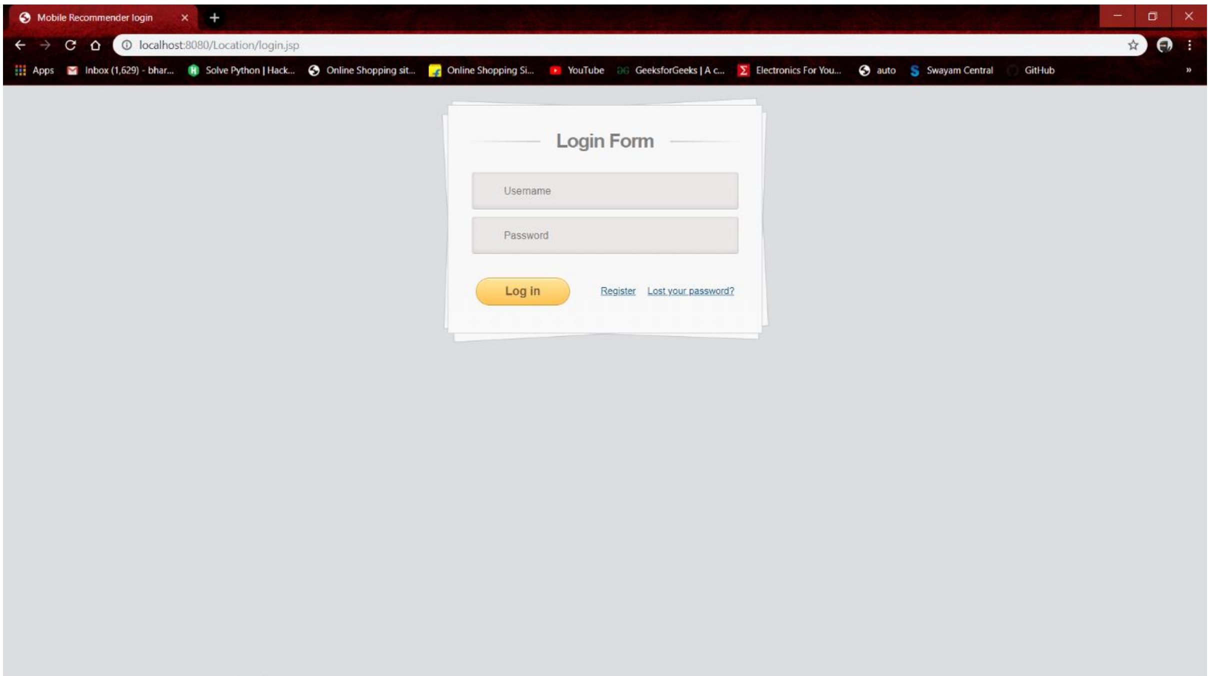
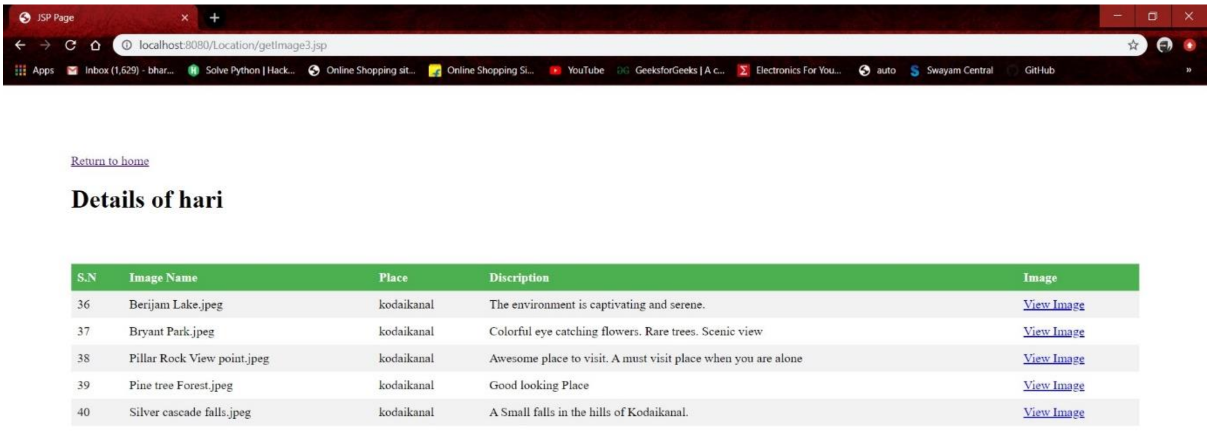



Fig.A2.2. Login page

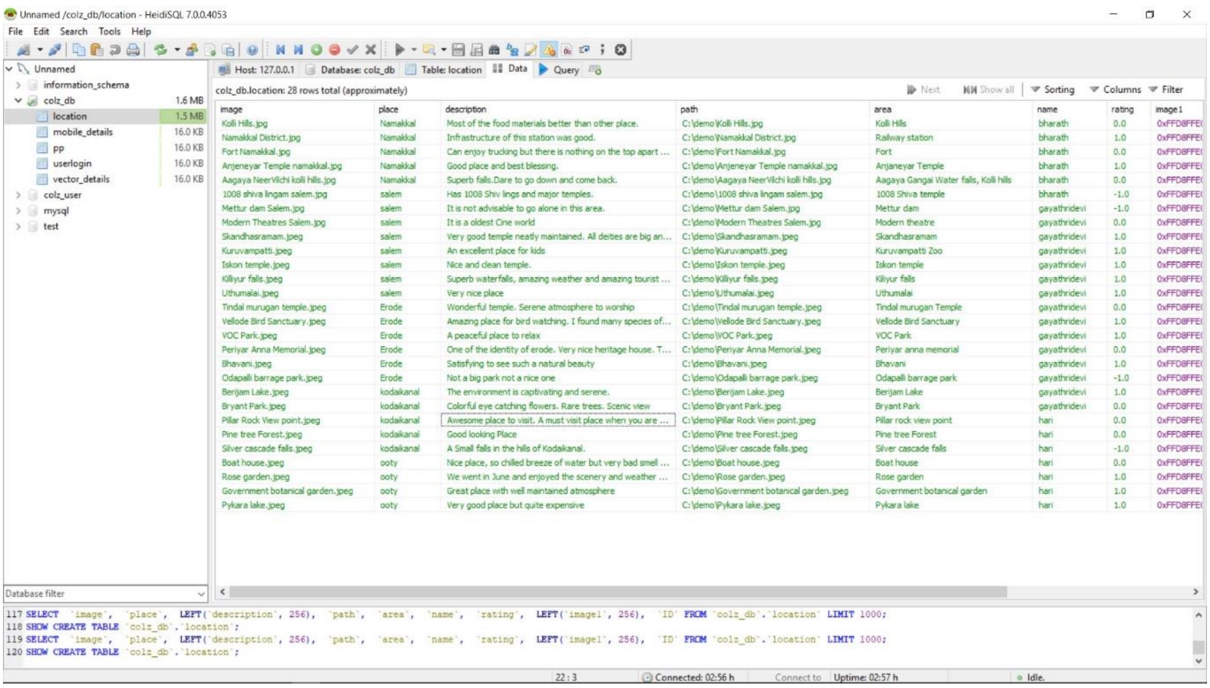


Return to home

Details of hari

| S.N | Image Name | Place | Discription | Image |
|---|---|---|---|---|
| 36 | Berijam Lake.jpeg | kodaikanal | The environment is captivating and serene. | View Image |
| 37 | Bryant Park.jpeg | kodaikanal | Colorful eye catching flowers. Rare trees. Scenic view | View Image |
| 38 | Pillar Rock View point.jpeg | kodaikanal | Awesome place to visit. A must visit place when you are alone | View Image |
| 39 | Pine tree Forest.jpeg | kodaikanal | Good looking Place | View Image |
| 40 | Silver cascade falls.jpeg | kodaikanal | A Small falls in the hills of Kodaikanal. | View Image |

Fig. A2.3. NLP based location recommendation



colz_db.location: 28 rows total (approximately)

| image | place | description | path | area | name | rating | image1 |
|---|---|---|---|---|---|---|---|
| Kolli Hills.jpg | Namakkal | Most of the food materials better than other place. | C:\demo\Kolli Hills.jpg | Kolli Hills | bharath | 0.0 | 0xFFD8FFE |
| Namakkal District.jpg | Namakkal | Infrastructure of this station was good. | C:\demo\Namakkal District.jpg | Railway station | bharath | 1.0 | 0xFFD8FFE |
| Fort Namakkal.jpg | Namakkal | Can enjoy trucking but there is nothing on the top apart ... | C:\demo\Fort Namakkal.jpg | Fort | bharath | 0.0 | 0xFFD8FFE |
| Anjeneyar Temple namakkal.jpg | Namakkal | Good place and best blessing. | C:\demo\Anjeneyar Temple namakkal.jpg | Anjaneyar Temple | bharath | 1.0 | 0xFFD8FFE |
| Aagaya NeerVilchi kolli hills.jpg | Namakkal | Superb falls.Dare to go down and come back. | C:\demo\Aagaya NeerVilchi kolli hills.jpg | Aagaya Gangai Water falls, Kolli hills | bharath | 0.0 | 0xFFD8FFE |
| 1008 shiva lingam salem.jpg | salem | Has 1008 Shiv lings and major temples. | C:\demo\1008 shiva lingam salem.jpg | 1008 Shiva temple | bharath | -1.0 | 0xFFD8FFE |
| Mettur dam Salem.jpg | salem | It is not advisable to go alone in this area. | C:\demo\Mettur dam Salem.jpg | Mettur dam | gayathridevi | -1.0 | 0xFFD8FFE |
| Modern Theatres Salem.jpg | salem | It is a oldest Cine world | C:\demo\Modern Theatres Salem.jpg | Modern theatre | gayathridevi | 0.0 | 0xFFD8FFE |
| Skandhasramam.jpeg | salem | Very good temple neatly maintained. All deities are big an... | C:\demo\Skandhasramam.jpeg | Skandhasramam | gayathridevi | 1.0 | 0xFFD8FFE |
| Kuruvampatti.jpeg | salem | An excellent place for kids | C:\demo\Kuruvampatti.jpeg | Kuruvampatti Zoo | gayathridevi | 1.0 | 0xFFD8FFE |
| Iskon temple.jpeg | salem | Nice and clean temple. | C:\demo\Iskon temple.jpeg | Iskon temple | gayathridevi | 1.0 | 0xFFD8FFE |
| Killiyur falls.jpeg | salem | Superb waterfalls, amazing weather and amazing tourist ... | C:\demo\Killiyur falls.jpeg | Kiliyur falls | gayathridevi | 1.0 | 0xFFD8FFE |
| Uthumalai.jpeg | salem | Very nice place | C:\demo\Uthumalai.jpeg | Uthumalai | gayathridevi | 1.0 | 0xFFD8FFE |
| Tindal murugan temple.jpeg | Erode | Wonderful temple. Serene atmosphere to worship | C:\demo\Tindal murugan temple.jpeg | Tindal murugan Temple | gayathridevi | 0.0 | 0xFFD8FFE |
| Vellode Bird Sanctuary.jpeg | Erode | Amazing place for bird watching. I found many species of... | C:\demo\Vellode Bird Sanctuary.jpeg | Vellode Bird Sanctuary | gayathridevi | 1.0 | 0xFFD8FFE |
| VOC Park.jpeg | Erode | A peaceful place to relax | C:\demo\VOC Park.jpeg | VOC Park | gayathridevi | 1.0 | 0xFFD8FFE |
| Periyar Anna Memorial.jpeg | Erode | One of the identity of erode. Very nice heritage house. T... | C:\demo\Periyar Anna Memorial.jpeg | Periyar anna memorial | gayathridevi | 0.0 | 0xFFD8FFE |
| Bhavani.jpeg | Erode | Satisfying to see such a natural beauty | C:\demo\Bhavani.jpeg | Bhavani | gayathridevi | 1.0 | 0xFFD8FFE |
| Odapalli barrage park.jpeg | Erode | Not a big park not a nice one | C:\demo\Odapalli barrage park.jpeg | Odapalli barrage park | gayathridevi | -1.0 | 0xFFD8FFE |
| Berijam Lake.jpeg | kodaikanal | The environment is captivating and serene. | C:\demo\Berijam Lake.jpeg | Berijam Lake | gayathridevi | 1.0 | 0xFFD8FFE |
| Bryant Park.jpeg | kodaikanal | Colorful eye catching flowers. Rare trees. Scenic view | C:\demo\Bryant Park.jpeg | Bryant Park | gayathridevi | 0.0 | 0xFFD8FFE |
| Pillar Rock View point.jpeg | kodaikanal | Awesome place to visit. A must visit place when you are ... | C:\demo\Pillar Rock View point.jpeg | Pillar rock view point | hari | 0.0 | 0xFFD8FFE |
| Pine tree Forest.jpeg | kodaikanal | Good looking Place | C:\demo\Pine tree Forest.jpeg | Pine tree Forest | hari | 0.0 | 0xFFD8FFE |
| Silver cascade falls.jpeg | kodaikanal | A Small falls in the hills of Kodaikanal. | C:\demo\Silver cascade falls.jpeg | Silver cascade falls | hari | -1.0 | 0xFFD8FFE |
| Boat house.jpeg | ooty | Nice place, so chilled breeze of water but very bad smell ... | C:\demo\Boat house.jpeg | Boat house | hari | 0.0 | 0xFFD8FFE |
| Rose garden.jpeg | ooty | We went in June and enjoyed the scenery and weather ... | C:\demo\Rose garden.jpeg | Rose garden | hari | 1.0 | 0xFFD8FFE |
| Government botanical garden.jpeg | ooty | Great place with well maintained atmosphere | C:\demo\Government botanical garden.jpeg | Government botanical garden | hari | 1.0 | 0xFFD8FFE |
| Pykara lake.jpeg | ooty | Very good place but quite expensive | C:\demo\Pykara lake.jpeg | Pykara lake | hari | 1.0 | 0xFFD8FFE |

Fig. A2.4. Database

## REFERENCES

[1] D. Lian, Y. Ge, F. Zhang, N. J. Yuan, X. Xie, T. Zhou, and Y. Rui [2015], "Content-aware collaborative filtering for location recommendation based on human mobility data," in Proceedings of ICDM'15. IEEE, pp. 261–270.

[2] V. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang [2010], "Collaborative filtering meets mobile recommendation: A user-centered approach," in Proceedings of AAAI'10. AAAl Press.

[3] M. Ye, P. Yin, W.-C. Lee, and D.-L. Lee [2011], "Exploiting geographical influence for collaborative point-of-interest recommendation," in Proceedings of SIGIR'11. ACM, pp. 325–334.

[4] W.-Y. Zhu, W.-C. Peng, L.-J. Chen, K. Zheng, and X. Zhou [2015], "Modeling user mobility for location promotion in location-based social networks," in Proceedings of KDD'15. ACM, pp. 1573–1582.

[5] B. Liu, Y. Fu, Z. Yao, and H. Xiong [2013], "Learning geographical preferences for point-of-interest recommendation," in Proceedings of KDD'13. ACM, pp. 1043–1051.

[6] H. Gao, J. Tang, X. Hu, and H. Liu [2013], "Exploring temporal effects for location recommendation on location-based social networks," in Proceedings of RecSys'13. ACM, pp. 93–100.

[7] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. M. Thalmann [2013], "Timeaware point-of-interest recommendation," in Proceedings of SIGIR' 13. ACM, pp. 363.

[8] Q. Yuan, G. Cong, and A. Sun [2014], “Graph-based point-of-interest recommendation with geographical and temporal influences,” in Proceedings of CIKM’14. ACM, pp. 659–668.

[9] A. Noulas, S. Scellato, N. Lathia, and C. Mascolo [2012], “A random walk around the city: New venue recommendation in location-based social networks,” in Proceedings of SocialCom’12. IEEE, pp. 144–153.

[10] P. Lops, M. De Gemmis, and G. Semeraro [2011], “Content-based recommender systems: State of the art and trends,” in Recommender systems handbook. Springer, pp. 73–105.

[11] D. Yang, D. Zhang, Z. Yu, and Z. Wang [2013], “A sentiment-enhanced personalized location recommendation system,” in Proceedings of HT’13. ACM, pp. 119–128.

[12] B. Liu and H. Xiong [2013], “Point-of-interest recommendation in location based social networks with topic and location awareness,” in Proceedings of SDM’13. SIAM, pp. 396–404.

[13] D. Lian, C. Zhao, X. Xie, G. Sun, E. Chen, and Y. Rui [2014], “Geomf: joint geographical modeling and matrix factorization for point-ofinterest recommendation,” in Proceedings of KDD’14. ACM, pp. 831–840.

[14] C. Cheng, H. Yang, I. King, and M. Lyu [2012], “Fused matrix factorization with geographical and social influence in location-based social networks,” in Proceedings of AAAI’12.

[15] Y. Liu, W. Wei, A. Sun, and C. Miao [2014], "Exploiting geographical neighborhood characteristics for location recommendation," in Proceedings of CIKM'14. ACM, pp. 739–748.

[16] C. R. Cloninger, T. R. Przybeck, and D. M. Svrakic [1994], The Temperament and Character Inventory (TCI): A guide to its development and use. center for psychobiology of personality, Washington University St. Louis, MO.

[17] S. Rendle [2012], "Factorization machines with libfm," ACM Transactions on Intelligent Systems and Technology (TIST), vol. 3, no. 3, p. 57.

[18] T. Chen, W. Zhang, Q. Lu, K. Chen, Z. Zheng, and Y. Yu [2012], "Svdfeature: a toolkit for feature-based collaborative filtering," Journal of Machine Learning Research, vol. 13, no. 1, pp. 3619–3622.

[19] D. Agarwal and B.-C. Chen [2009], "Regression-based latent factor models," in Proceedings of KDD'09. ACM, pp. 19–28.

[20] D. H. Stern, R. Herbrich, and T. Graepel [2009], "Matchbox: large scale online bayesian recommendations," in Proceedings of WWW'09. ACM, pp. 111–120.