IMPROVED RANKING BASED RECOMMENDATION SYSTEM BY FUZZY DATA ANALYSIS

Project report submitted in partial fulfilment of the Requirements for the Award of the Degree of B.Tech in Computer Science and Engineering

BY

ROOPAL JOSHI

2006336

TANMAYA BHATT

2006374

TITIKSHA NAITHANI

2006378

Under the Guidance of

Mr. Vijay Singh

Assistant Professor (CSE)

Department of Computer Science and Engineering

Graphic Era University

(Under Section 3 of UGC Act, 1956)

Dehradun-248002 2016

IMPROVED RANKING BASED RECOMMENDATION SYSTEM BY FUZZY DATA ANALYSIS

Project report submitted in partial fulfilment of the

Requirements for the Award of the

Degree of B.Tech in

Computer Science and Engineering

BY

ROOPAL JOSHI

2006336

TANMAYA BHATT

2006374

TITIKSHA NAITHANI

2006378

Under the Guidance of

Mr. Vijay Singh

Assistant Professor (CSE)

Department of Computer Science and Engineering

Graphic Era University

(Under Section 3 of UGC Act, 1956)

Dehradun-248002 2016

CERTIFICATE

This is to certify that the project report entitled IMPROVED RANKING BASED RECOMMENDATION SYSTEM BY FUZZY DATA ANALYSIS being submitted by

Roopal Joshi - 2006336

Tanmaya Bhatt - 2006374

Titiksha Naithani -2006378

partial fulfilment for the award of the Degree of Bachelor of Technology in Computer Science and Engineering to the Graphic Era University is a record of bona fide work carried out under my guidance and supervision.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

(Mr. Vijay Singh)

Assistant Professor

Date:

(Mr. Dibyahash Bordoloi)

Head of Departement, CSE

ACKNOWLEDGEMENT

The completion of project brings with it a satisfaction, but it is never complete without thanking those people who made it possible and whose constant support crowned our efforts with success.

We would like to express our gratitude to Mr. Dibyahash Bordoloi, Head of the Department, Computer Science and engineering, GEU, Dehradun for encouraging and inspiring us to carry out the project in the department lab.

We would also like to thanks our guide Mr. Vijay Singh, Department of Science and engineering for their expert guidance, encouragement and valuable suggestions at every step.

We would like to thanks all the staff members of CSE department for providing us with the required facilities and support towards the completion of the project.

Date:

Place: (ROOPAL JOSHI)

(TANMAYA BHATT)

(TITIKSHA NAITHANI)

Abstract

Recommender system is a personalized information filtering technique used to identify desired number of items based on interests of a user. It is based on past behaviour , relations to other users , item similarity and context. A system is developed using past user ratings by applying different techniques. The techniques focus on improving the accuracy of the recommendations and other important aspects of recommendation quality, such as the diversity of recommendations. Fuzzy systems are used to solve the data analysis problems with the main aim of creating models for predictions and interpretation. In this paper we aim to improve recommendations by fuzzifying a TripAdvisor dataset.

We summed all dataset as total sum of each character like room service etc for each we found the percentage value of the data, for which we calculated the total possible value by multiplying number of reviews and value 5.

On this percentage value we applied the triangular membership function for fuzzification of the data so that we could note down the accuracy. We executed the data in libSVM.

We started the execution by making libSVM training classes then we trained the data by using svm_train function and got trained files. In the next step we tested the data with the originally given data by using svm_test function . After that we noted the accuracy of the data.

In the end we gave the accuracy to prove that recommendations are better using libSVM.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Recommender systems has revolutionized the way people find products, information, and even other people online. These systems study and find patterns of different behaviour to know what someone will prefer from among a collection of things that , that person has never experienced. The technology behind recommender systems has evolved over the past few years into a humongous collection of tools that help enable the researcher to develop effective algorithms for better accuracy of recommendation systems.

The algorithms are differentiated usually by the type of filter they have. There are different approaches to recommendation like

Collaborative Filtering(Items recommended based only on the user's past behavior), Content Based(Items recommended based on item features),Personalized Learning To Rank (Recommendation is treated as a ranking problems),Demographic (Items recommended based on user features), Social Recommendations(trust based), Hybrid(combine any two of the above).

Traditionally, recommendation systems gather the information by explicitly asking the users or by implicitly collecting the information by their behaviour.But sometimes users don't give the correct information about them. So the alternative of this problem is collecting the information about the user preferences from their behaviour which can be found by their recent activities which are available in online communities.

In this paper we report on the method by which we can create recommendation system which will work by using fuzzy data analysis.To give better recommendations we have used trip advisor data set in this paper where we have reviews of the hotels given by actual visitors to these hotels.

1.2 Recommendation System

Recommendation systems (sometimes replacing "system" with a synonym such as platform or engine) are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that a user would give to an item.It is a system that produces individualized recommendations as output or has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible options.

Recommender systems have become extremely common in recent years, and are applied in a variety of applications. The most popular ones are probably movies, music, news, books, research articles, search queries, social tags, and products in general. However, there are also recommender systems for experts, collaborators, jokes, restaurants, financial services, life insurance, persons (online dating), and Twitter followers.

1.3 Applications of Recommendation Systems

Product Recommendations: Perhaps the most important use of recommendation systems is at on-line retailers. These suggestions are not random, but are based on the purchasing decisions made by similar customers or on other techniques

Movie Recommendations: Netflix offers its customers recommendations of movies they might like. These recommendations are based on ratings provided by users,

News Articles: News services have attempted to identify articles of interest to readers, based on the articles that they have read in the past. The similarity might be based on the similarity of important words in the documents, or on the articles that are read by people with similar reading tastes. The same principles apply to recommending blogs from among the millions of blogs available, videos on YouTube, or other sites where content is provided regularly.

1.4 Types of Recommender Systems

Recommendation systems use a number of different technologies. These can be classified into two broad groups.

Content-based systems examine properties of the items recommended. For instance, if a Netflix user has watched many cowboy movies, then recommend a movie classified in the database as having the “cowboy” genre.

Collaborative filtering systems recommend items based on similarity measures between users and/or items. The items recommended to a user are those preferred by similar users

The differences between collaborative and content-based filtering can be demonstrated by comparing two popular music recommender systems – Last.fm and Pandora Radio.

Last.fm creates a "station" of recommended songs by observing what bands and individual tracks the user has listened to on a regular basis and comparing those against the listening behaviour of other users. Last.fm will play tracks that do not appear in the user's library, but are often played by other users with similar interests. As this approach leverages the behaviour of users, it is an example of a collaborative filtering technique.

Pandora uses the properties of a song or artist (a subset of the 400 attributes provided by the Music Genome Project) in order to seed a "station" that plays music with similar properties. User feedback is used to refine the station's results, deemphasizing certain attributes when a user "dislikes" a particular song and emphasizing other attributes when a user "likes" a song. This is an example of a content-based approach.

Each type of system has its own strengths and weaknesses. In the above example, Last.fm requires a large amount of information on a user in order to make accurate recommendations. This is an example of the cold start problem, and is common in collaborative filtering systems. While Pandora needs very little information to get started, it is far more limited in scope (for example, it can only make recommendations that are similar to the original seed).

Recommender systems are a useful alternative to search algorithms since they help users discover items they might not have found by themselves. Interestingly enough, recommender systems are often implemented using search engines indexing non-traditional data.

1.5 Fuzzy data analysis

1.5.1 Fuzzy Logic

Fuzzy logic is a form of many-valued logic in which the truth values of variables may be any real number between 0 and 1, considered to be "fuzzy". By contrast, in Boolean logic, the truth values of variables may only be 0 or 1, often called "crisp" values. Fuzzy logic has been employed to handle the concept of partial truth, where the truth value may range between completely true and completely false. Furthermore, when linguistic variables are used, these degrees may be managed by specific (membership) functions. Humans and animals often operate using fuzzy evaluations in many everyday situations. In the case where someone is tossing an object into a container from a distance, the person does not compute exact values for the object weight, density, distance, direction, container height and width, and air resistance to determine the force and angle to toss the object. Instead the person instinctively applies quick “fuzzy” estimates, based upon previous experience, to determine what output values of force, direction and vertical angle to use to make the toss.

Both degrees of truth and probabilities range between 0 and 1 and hence may seem similar at first. For example, let a 100 ml glass contain 30 ml of water. Then we may consider two concepts: empty and full. The meaning of each of them can be represented by a certain fuzzy set. Then one might define the glass as being 0.7 empty and 0.3 full. Note that the concept of emptiness would be subjective and thus would depend on the observer or designer. Another designer might, equally well, design a set membership function where the glass would be considered full for all values down to 50 ml. It is essential to realize that fuzzy logic uses degrees of truth as a mathematical model of vagueness, while probability is a mathematical model of ignorance.

The term fuzzy logic was introduced with the 1965 proposal of fuzzy set theory by LotfiZadeh. Fuzzy logic had however been studied since the 1920s, as infinite-valued logic—notably by Łukasiewicz and Tarski.

Fuzzy logic has been applied to many fields, from control theory to artificial intelligence.

Humans and animals often operate using fuzzy evaluations in many everyday situations. In the case where someone is tossing an object into a container from a distance, the person does not compute exact values for the object weight, density, distance, direction, container height and width, and air resistance to determine the force and angle to toss the object. Instead the person instinctively applies quick "fuzzy" estimates, based upon previous experience, to determine what output values of force, direction and vertical angle to use to make the toss.

Both degrees of truth and probabilities range between 0 and 1 and hence may seem similar at first. For example, let a 100 ml glass contain 30 ml of water. Then we may consider two concepts: empty and full. The meaning of each of them can be represented by a certain fuzzy set. Then one might define the glass as being 0.7 empty and 0.3 full. Note that the concept of emptiness would be subjective and thus would depend on the observer or designer. Another designer might, equally well, design a set membership function where the glass would be considered full for all values down to 50 ml. It is essential to realize that fuzzy logic uses degrees of truth as a mathematical model of vagueness, while probability is a mathematical model of ignorance.

1.5.2 Fuzzy set

In mathematics, fuzzy sets are sets whose elements have degrees of membership. Fuzzy sets were introduced by Lotfi A. Zadeh and Dieter Klaua in 1965 as an extension of the classical notion of set. At the same time, Salii (1965) defined a more general kind of structure called an L-relation, which he studied in an abstract algebraic context. Fuzzy relations, which are used now in different areas, such as linguistics (De Cock, Bodenhofer&Kerre 2000) decision-making (Kuzmin 1982) and clustering (Bezdek 1978), are special cases of L-relations when L is the unit interval.

A fuzzy set is a pair (U, m) where U is a set and $m: U \to [0, 1]$.

For each $x \in U$, the value $m(x)$ is called the grade of membership of x in (U, m). For a finite set $U = \{x_1, \ldots, x_n\}$, the fuzzy set (U, m) is often denoted by $\{m(x_1)/x_1, \ldots, m(x_n)/x_n\}$.

Let $x \in U$. Then x is called not included in the fuzzy set (U, m) if $m(x) = 0$, x is called fully included if $m(x) = 1$, and x is called a fuzzy member if $0 < m(x) < 1$.[5] The set $\{x \in U \mid m(x) > 0\}$ is called the support of (U, m) and the set $\{x \in U \mid m(x) = 1\}$ is called its kernel or core. The function m is called the membership function of the fuzzy set (U, m).

Sometimes, more general variants of the notion of fuzzy set are used, with membership functions taking values in a (fixed or variable) algebra or structure L of a given kind; usually it is required that L be at least a lattice. These are usually called L-fuzzy sets, to distinguish them from those valued over the unit interval. The usual membership functions with values in [0, 1] are then called [0, 1]-valued membership functions.

1.5.3 Fuzzy clustering

Fuzzy clustering (also referred to as soft clustering) is a form of clustering in which each data point can belong to more than one cluster or partition.

Clustering or cluster analysis involves assigning data points to clusters (also called buckets, bins, or classes), or homogeneous classes, such that items in the same class or cluster are as similar as possible, while items belonging to different classes are as dissimilar as possible. Clusters are identified via similarity measures. These similarity measures include distance, connectivity, and intensity. Different similarity measures may be chosen based on the data or the application.

1.5.4 Fuzzy Database

If a regular or classical database is a structured collection of information (records or data) stored in a computer, a fuzzy database is a database which is able to deal with uncertain or incomplete information using fuzzy logic. There are many forms of adding flexibility in fuzzy databases. The simplest technique is to add a fuzzy membership degree to each record, that is, an attribute in the range [0,1]. However, there are other kinds of databases allowing fuzzy values to be stored in fuzzy attributes using fuzzy sets, possibility distributions, or fuzzy degrees associated to some attributes and with different meanings (membership degree, importance degree, fulfilment degree, etc.). Of course, fuzzy databases should allow fuzzy queries using fuzzy or non fuzzy data and there are some languages that allow this kind of queries, like FSQL or SQLf. In synthesis, the research in fuzzy databases includes the following areas: flexible querying in classical or fuzzy databases, extending classical data models in order to achieve fuzzy databases (fuzzy relational databases, fuzzy object-oriented databases, etc.), fuzzy conceptual modelling, fuzzy data mining techniques, and applications of these advances in real databases.

Fuzzy databases intend to grasp imperfect information about a modelled part of the world and represent it directly, as accurate as possible, in a database. The two leading approaches to the representation of imperfect information in databases are the possible approach and the similarity relation based approach.

Once fuzzy relations are defined, it is possible to develop fuzzy relational databases. The first fuzzy relational database, FRDB, appeared in Maria Zemankova's dissertation. Later, some other models arose like the Buckles-Petry model, the Prade-Testemale Model, the Umano-Fukami model or the GEFRED model by J.M. Medina, M.A. Vila et al. In the context of fuzzy databases, some fuzzy querying languages have been defined, highlighting the SQLf by P. Bosc et al. and the FSQL by J. Galindo et al.

CHAPTER 2

LITERATURE SURVEY

Recommender systems or recommendation systems (sometimes replacing "system" with a synonym such as platform or engine) are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that a user would give to an item.

Recommender systems have become extremely common in recent years, and are applied in a variety of applications. The most popular ones are probably movies, music, news, books, research articles, search queries, social tags, and products in general. However, there are also recommender systems for experts, collaborators, jokes, restaurants, financial services,[6] life insurance, persons (online dating), and Twitter followers.

Recommender systems typically produce a list of recommendations in one of two ways – through collaborative or content-based filtering. Collaborative filtering approaches building a model from a user's past behaviour (items previously purchased or selected and/or numerical ratings given to those items) as well as similar decisions made by other users. This model is then used to predict items (or ratings for items) that the user may have an interest in.

Each type of system has its own strengths and weaknesses

Recommender systems are a useful alternative to search algorithms since they help users discover items they might not have found by themselves. Interestingly enough, recommender systems are often implemented using search engines indexing non-traditional data.

Montaner provides the first overview of recommender systems, from an intelligent agents perspective

Collaborative filtering

Collaborative filtering (CF) is a technique used by some recommender systems. Collaborative filtering has two senses, a narrow one and a more general one. In general, collaborative filtering is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc. Applications of collaborative filtering typically involve very large data sets. Collaborative filtering methods have been applied to many different kinds of data including: sensing and monitoring data, such as in mineral exploration, environmental sensing over large areas or multiple sensors; financial data, such as financial service institutions that integrate many financial sources; or in electronic commerce and web applications where the focus is on user data, etc. The remainder of this discussion focuses on collaborative filtering for user data, although some of the methods and approaches may apply to the other major applications as well.

In the newer, narrower sense, collaborative filtering is a method of making automatic predictions (filtering) about the interests of a user by collecting preferences or taste information from many users (collaborating). The underlying assumption of the collaborative filtering approach is that if a person *A* has the same opinion as a person *B* on an issue, A is more likely to have B's opinion on a different issue *x* than to have the opinion on x of a person chosen randomly. For example, a collaborative filtering recommendation system for television tastes could make predictions about which television show a user should like given a partial list of that user's tastes (likes or dislikes). Note that these predictions are specific to the user, but use information gleaned from many users. This differs from the simpler approach of giving an average (non-specific) score for each item of interest, for example based on its number of votes.

Content-based filtering

Another common approach when designing recommender systems is content-based filtering. Content-based filtering methods are based on a description of the item and a profile of the user's preference.[In a content-based recommender system, keywords are used to describe the items and a user profile is built to indicate the type of item this user likes. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past (or is examining in the present). In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. This approach has its roots in information retrieval andin formation filtering research.

To abstract the features of the items in the system, an item presentation algorithm is applied. A widely used algorithm is the tf–idf representation (also called vector space representation).

To create a user profile, the system mostly focuses on two types of information: 1. A model of the user's preference. 2. A history of the user's interaction with the recommender system.

Fuzzy logic is a form of many-valued logic in which the truth values of variables may be any real number between 0 and 1, considered to be "fuzzy". By contrast, in Boolean logic, the truth values of variables may only be 0 or 1, often called "crisp" values. Fuzzy logic has been employed to handle the concept of partial truth, where the truth value may range between completely true and completely false.

LIBSVM

Main features of LIBSVM include

Different SVM formulations

Efficient multi-class classification

Cross validation for model selection

Probability estimates

Various kernels (including precomputed kernel matrix)

Weighted SVM for unbalanced data

Both C++ and Java sources

GUI demonstrating SVM classification and regression

Python, R, MATLAB, Perl, Ruby, Weka, Common
LISP, CLISP, Haskell, OCaml, LabVIEW, and PHP interfaces. C# .NET code
and CUDA extension is available.
It's also included in some data mining environments: RapidMiner, PCP,
and LIONsolver.

Automatic model selection which can generate contour of cross validation accuracy.

some other useful programs in this package.

svm-scale:

This is a tool for scaling input data file.

svm-toy:

This is a simple graphical interface which shows how SVM separate data in a plane. You can click in the window to draw data points. Use "change" button to choose class 1, 2 or 3 (i.e., up to three classes are supported), "load" button to load data from a file, "save" button to save data to a file, "run" button to obtain an SVM model, and "clear" button to clear the window.

You can enter options in the bottom of the window, the syntax of options is the same as `svm-train'.

svm-train' Usage

=================

Usage: svm-train [options] training_set_file [model_file]

options:

-s svm_type : set type of SVM (default 0)

0 -- C-SVC

1 -- nu-SVC

2 -- one-class SVM

3 -- epsilon-SVR

4 -- nu-SVR

-t kernel_type : set type of kernel function (default 2)

0 -- linear: u'*v

1 -- polynomial: (gamma*u'*v + coef0)^degree

2 -- radial basis function: exp(-gamma*|u-v|^2)

3 -- sigmoid: tanh(gamma*u'*v + coef0)

-d degree : set degree in kernel function (default 3)

-g gamma : set gamma in kernel function (default 1/k)

-r coef0 : set coef0 in kernel function (default 0)

-c cost : set the parameter C of C-SVC, epsilon-SVR, and nu-SVR (default 1)

-n nu : set the parameter nu of nu-SVC, one-class SVM, and nu-SVR (default 0.5)

-p epsilon : set the epsilon in loss function of epsilon-SVR (default 0.1)

-m cachesize : set cache memory size in MB (default 40)

-e epsilon : set tolerance of termination criterion (default 0.001)

-h shrinking: whether to use the shrinking heuristics, 0 or 1 (default 1)

-b probability_estimates: whether to train an SVC or SVR model for probability estimates, 0 or 1 (default 0)

-wi weight: set the parameter C of class i to weight*C in C-SVC (default 1)

-v n: n-fold cross validation mode

The k in the -g option means the number of attributes in the input data.

option -v randomly splits the data into n parts and calculates cross

validation accuracy/mean squared error on them.

`svm-predict' Usage

===================

Usage: svm-predict [options] test_file model_file output_file

options:

-b probability_estimates: whether to predict probability estimates, 0 or 1 (default 0); one-class SVM not supported yet

model_file is the model file generated by svm-train.

test_file is the test data you want to predict.

CHAPTER 3

SOFTWARE REQUIRMENT ANALYSIS

3.1 Problem statement

Providing related content out of relevant and irrelevant collection of items to users of online service providers. When a user first enters into the recommendation system, the system knows nothing about user's preferences. Consequently, the system is unable to present any personalized recommendations. Mostly recommendations are inaccurate or improper , so our main aim is to make a better recommendation system.

The ever-increasing number of E-commerce sites on the Internet has brought about information overload. This has made it difficult for consumers of certain products to find information about such products in an attempt to purchase products that best satisfies them. It has equally reduced the volume of product sales in the E-commerce domain. This project proposes a personalized recommender system driven by fuzzy logic technique.

The large amount of product information on the Web poses great challenges to both customers and online businesses. More customers are turning towards online shopping because it is relatively convenient, reliable, and fast; yet such customers usually experience difficulty in searching for products on the Web due to information overload. Online businesses have often been overwhelmed by the rich data they have collected and find it difficult to promote products appropriate to specific customers. There is also the problem of ineffective utilization of the available large amount of product information from online transactions to support better decision making by both buyers and sellers.

A recommender system analyzes data about items, or interactions between users and items in order to find associations between items and users. It provides advice to users about items they might wish to purchase or examine. The recommendations made by such a system can help users navigate through large information spaces of product descriptions, news articles or other items . Various factors are considered when recommending products to online buyers; these include: top sellers of a particular product, demographic information of buyers, and analysis of the past buying behaviour of customers to predict their buying behaviours in the future. These forms of recommendation include suggesting products to the consumer, providing personalized product information, summarizing community opinions, and providing community critiques.

Recommendation systems enable consumers to easily access information about products they are interested in, and save time of reading through electronic documents. Moreover, enterprises can get to know customers' buying behaviours better, and develop efficient marketing strategies to attract different customers. Customer's satisfaction, and loyalty can thus be increased; the increase in the visiting frequency of customers can further create more transaction opportunities and benefit the Internet enterprises [12]. A good personalized recommendation system should be able to improve user satisfaction; a key attribute to customer loyalty and continued use.

3.2 Modules and their functionalities

3.2.1 TripAdvisor dataset

The TripAdvisor dataset consists of around 240,000 customer-supplied reviews of 1,850 hotels. Each review is associated with a hotel and a star-rating, 1-star (most negative) to 5-star (most positive), chosen by the customer to indicate his/her evaluation. This dataset contains around 90,000 hotel reviews, in three subsets(for libsvm): the train, validation and test subsets contain approximately 76,000, 6,000 and 13,000 reviews respectively. Each of these three subsets contains a balanced number of negative (1-star and 2-star) and positive (4-star and 5-star) reviews. The dataset also includes neutral reviews (e.g. with a rating value of 3) that are used in three-class classification. For binary classification, these neutral reviews are omitted from the dataset.

TripAdvisor datasets

2015-2016

Tripadvisor dataset: Includes personality scores (calculated using Fabio Celli's component).

DOI: 10.13140/RG.2.1.5104.8081

- detailed description of users' profiles
- personality scores per each user profile
- samples of 5 or more text reviews (for each user)
- textual content of 1 article (available only for some users)

Dataset description

Tripadvisor dataset: Updated version of TripAdvisor Datasets released in 2012-2013.

DOI: 10.13140/RG.2.1.3308.0409

- detailed description of users' profiles
- samples of 5 or more text reviews (for each user)
- textual content of 1 article (available only for some users)

History

In February 2000, TripAdvisor was founded by Stephen Kaufer, Langley Steinert, and several others. Kaufer says the original idea wasn't a user generated social media site to swap reviews. Rather, "We started as a site where we were focused more on those official words from guidebooks or newspapers or magazines. We also had a button in the very beginning that said, "Visitors add your own review", and boy, did that just take off. Pretty soon the number of average consumer reviews far surpassed the number of 'professional reviews'. That is when the site really turned into this collection of what the normal traveler was saying wherever they were going." Original financing was obtained from Flagship Ventures, the Bollard Group, and private investors.

In 2004, the company was purchased by IAC/InterActiveCorp.

In August 2005, IAC spun off its travel group of businesses under the Expedia, Inc. name.

In April 2009, TripAdvisor launched its official site in China, www.daodao.com. Since then it has indexed more than 20,000 hotels and restaurants information and customer reviews, and made top lists, becoming one of the biggest travel websites as of July 2011.

In September 2010, SmarterTravel, part of TripAdvisor Media Group, launched SniqueAway (now Jetsetter), the first members-only site where each travel deal is endorsed by the people.

In March 2011, TripAdvisor informed all registered TripAdvisor members that an unauthorized third party had stolen some of TripAdvisor's email list and might use it to create spam messages. No passwords or other information was stolen. This happened shortly before many other companies reported similar thefts of the addresses on their email lists.

In April 2011, it was announced that Expedia would split into two publicly traded companies by spinning off the TripAdvisor brand of travel sites. According to Expedia CEO Dara Khosrowshahi, the move "allows the two businesses to be pure plays and to operate with the proper amount of focus to grow respectively."

According to a July 2011 PhoCusWright survey of 3,641 respondents, solicited at random through a pop-up invitation link on TripAdvisor.com and commissioned by Trip Advisor, "98% of participants found that TripAdvisor's hotel reviews … accurately reflect the experience."

In December 2011, TripAdvisor was spun off from Expedia in a public offering.

TripAdvisor states it is the world's largest travel site, with nearly 280 million unique monthly visitors.

In April 2012, the company launched a connection to Facebook that lets users select reviews from people in their social graph.

In August 2014, a survey found that TripAdvisor was the most widely recognized, used, and trusted travel website.

In October 2014, Trip Advisor released a new feature, "Just for you", offering tailored hotel recommendations based on the user's preferences and search history on the site.

In 2015, Tripadvisor started let users not only rate places but tag them, saying, for example that a place propose brunch, is open on Sunday, offers delivery, takeout, etc.

3.2.2 Triangular membership

Fuzzifying Data means to convert a given data into fuzzy data.

A membership function characterizes the fuzzy set perfectly. A membership function has a membership value or its degree which tells about the grade of membership of the element of the function to the fuzzy set. It is used to graphically represent the fuzzy set. Such that in the graph, x axis is used to represents the universe of discourse, and the y axis is used o represent the degrees of the membership. The triangular membership function is specified by its three parameters. In which the curve is a function of a vector x, and depends on the scalar parameters a, b and c.

$$\mathrm{triangle}(x; a, b, c) = \begin{cases} 0, x \le a \\ \dfrac{x-a}{b-a}, a \le x \le b \\ \dfrac{c-x}{c-b}, b \le x \le c \\ 0, c \ge x \end{cases}$$

where a is the lower limit , c is the upper limit and b is the value.

We took each %value from above and fuzzified it. Each of these %values was subjected to the triangular membership function and was given it’s equivalent fuzzy value.

It is to be remembered that the triangular membership function gives data in ranges. Thus the values closer to the peak have a value nearing 1 while the values near the lower and upper limits have values nearing 0.

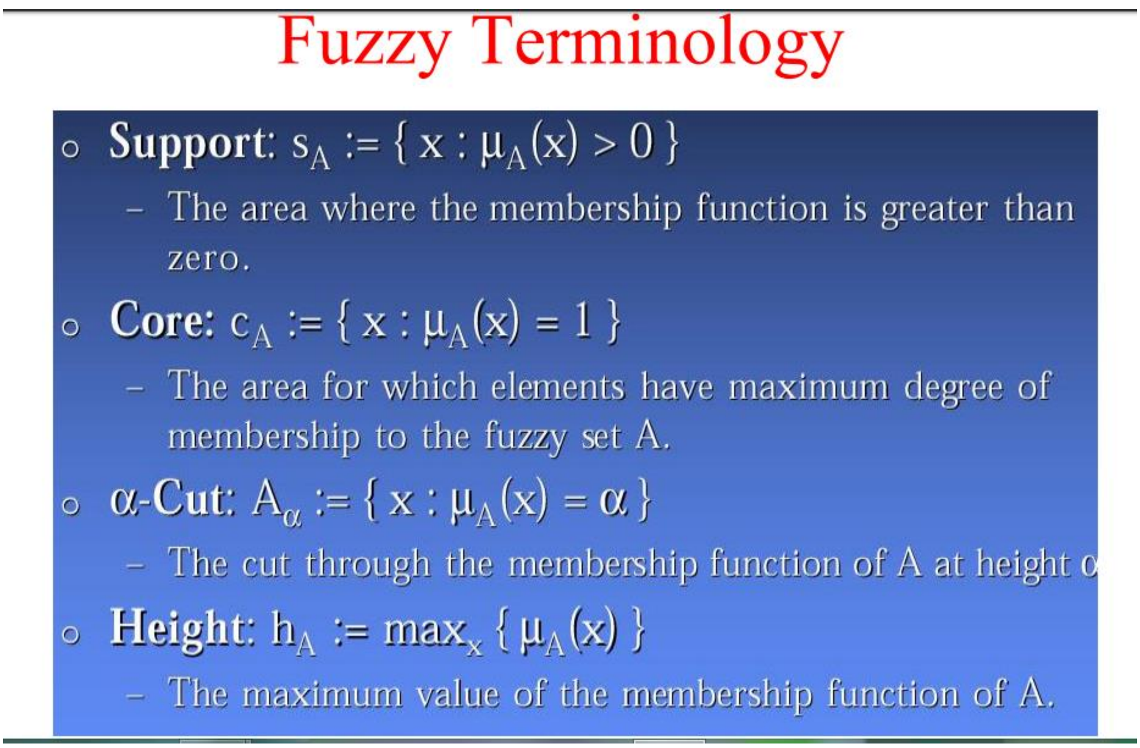


Fig 1

Input membership functions themselves can take any form the designer of the system requires triangles, trapezoids, bell curves or any other shape as long as those shapes accurately represent the distribution of information within the system, and as long as a region of transition exists between adjacent membership functions.

Example

Triangle-Shaped Membership Function

```
x = 0:0.1:10;
y = trimf(x,[3 6 8]);
plot(x,y)
xlabel('trimf, P = [3 6 8]')
```

ylim([-0.05 1.05])

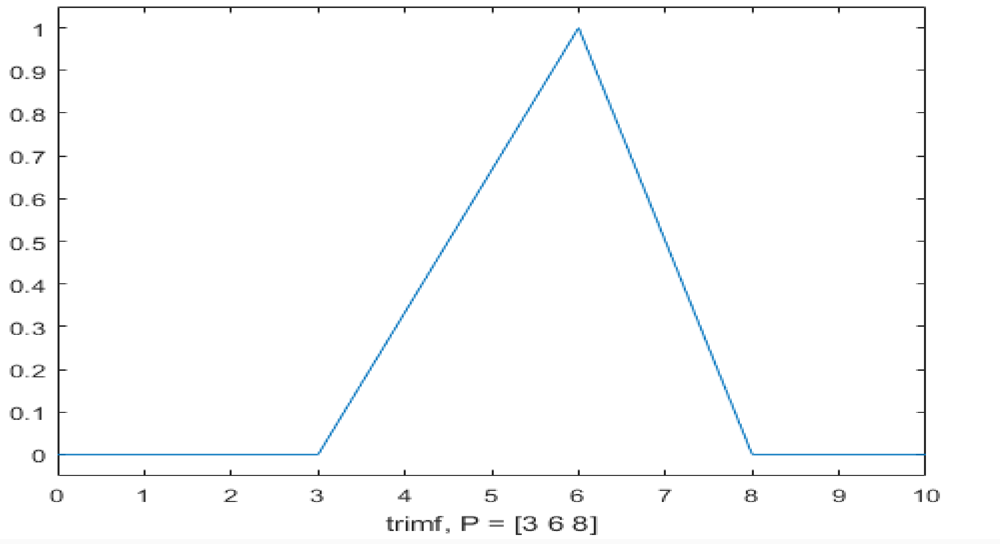


Fig 2

The most commonly used in practice are Triangles , Trapezoids, Bell curves, Gaussian, and Sigmoidal

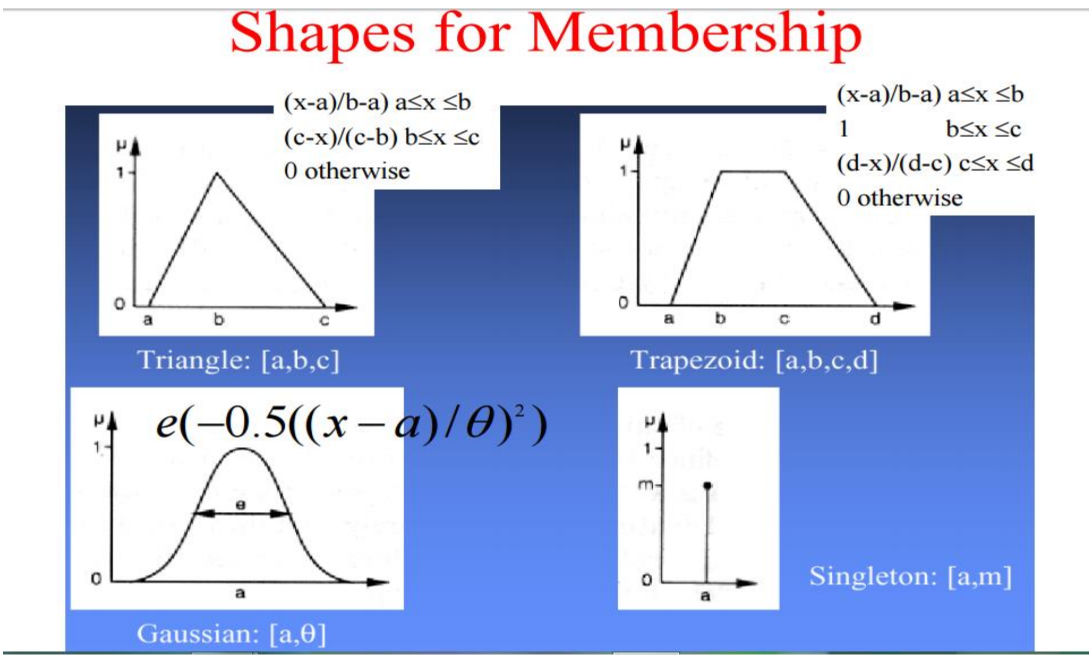


Fig 3

3.2.2.1Features of Membership Functions :-

Core of a Membership Function :-

Core of a membership function for a fuzzy set A is defined as that region of universe that is characterized by complete or full membership in the set A. Therefore core consists of all those elements X of universe of discourse, such that

$$\mu_A(x) = 1$$

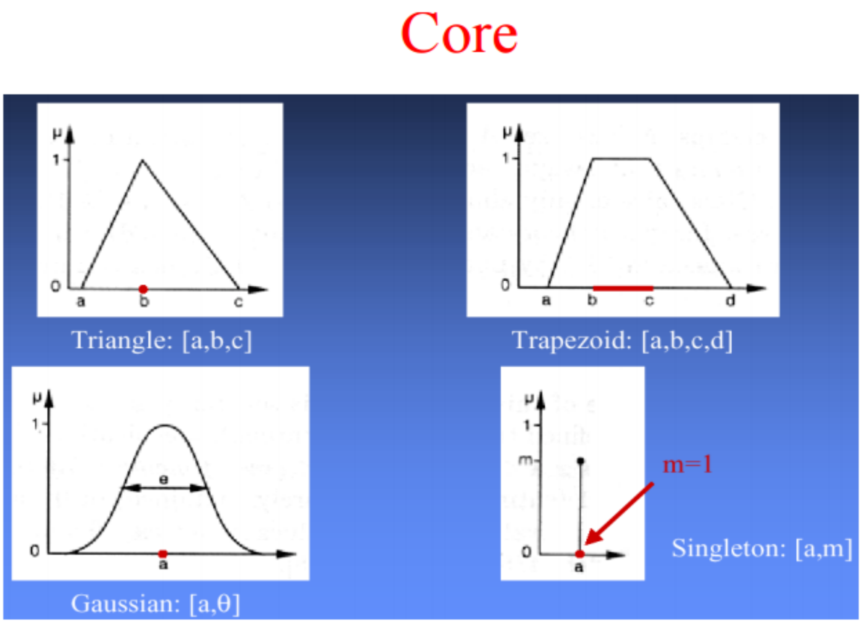


Fig 4

Support of a Membership Function

Support of a membership function for a fuzzy set A is defined as that region of universe that is characterized by non-zero membership in the fuzzy set A. So support consists of all those elements X of universe, such that

$$\mu_A(x) > 0$$

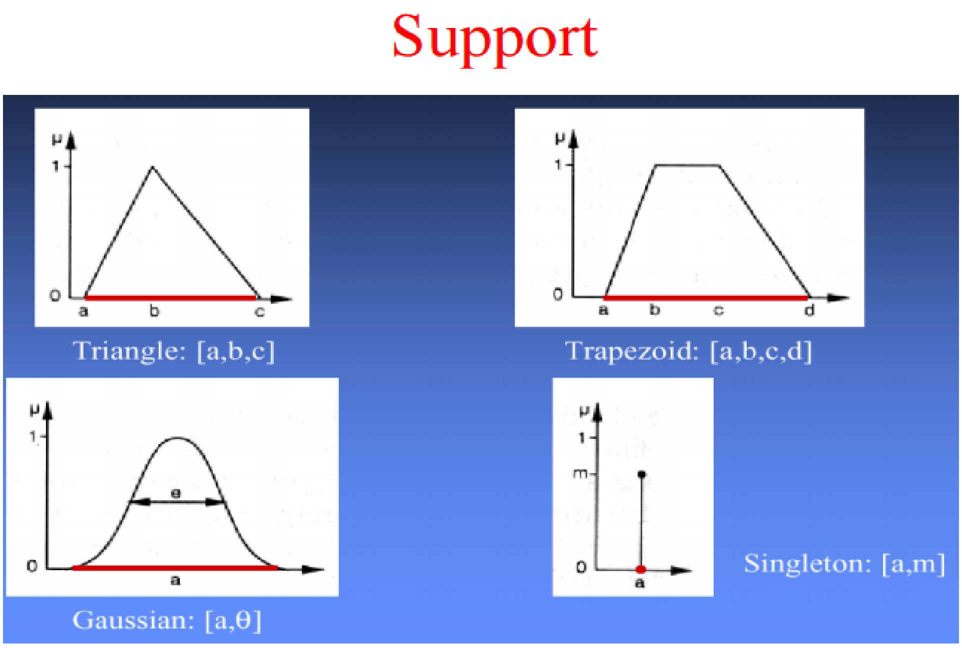


Fig 5

Boundary of a Membership Functions

Boundary of a membership function for a fuzzy set A is defined as that region of universe X, that is characterized by non-zero membership but not complete membership. Boundaries comprises that part of elements X of Universe of Discourse whose membership value is given by

$$\mu_A(x) \in (0,1)$$

Core, Support and Boundary of a membership function representation

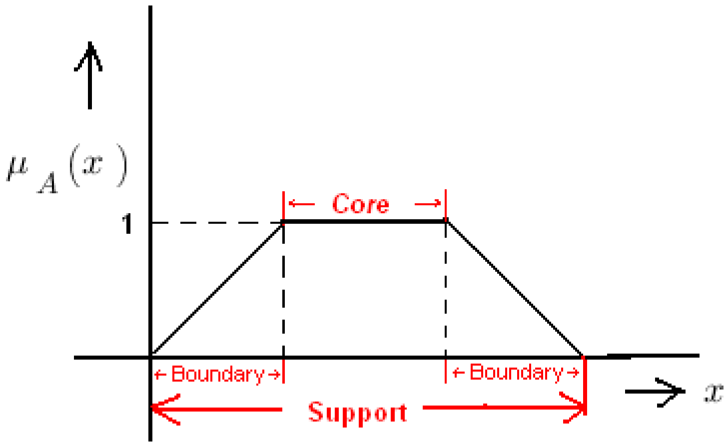


Fig 6

Cross-over Points of a Membership Function

It is defined as the elements of a fuzzy set A whose membership value is equal to 0.5

$$\mu_A(x) = 0.5$$

Height of a Membership Functions

Height of a membership function is the maximum value of the membership function. If the height of a fuzzy set is < 1 then it subnormal fuzzy set. Whereas if its height is equal to 1 then it is a normal fuzzy set.

Height

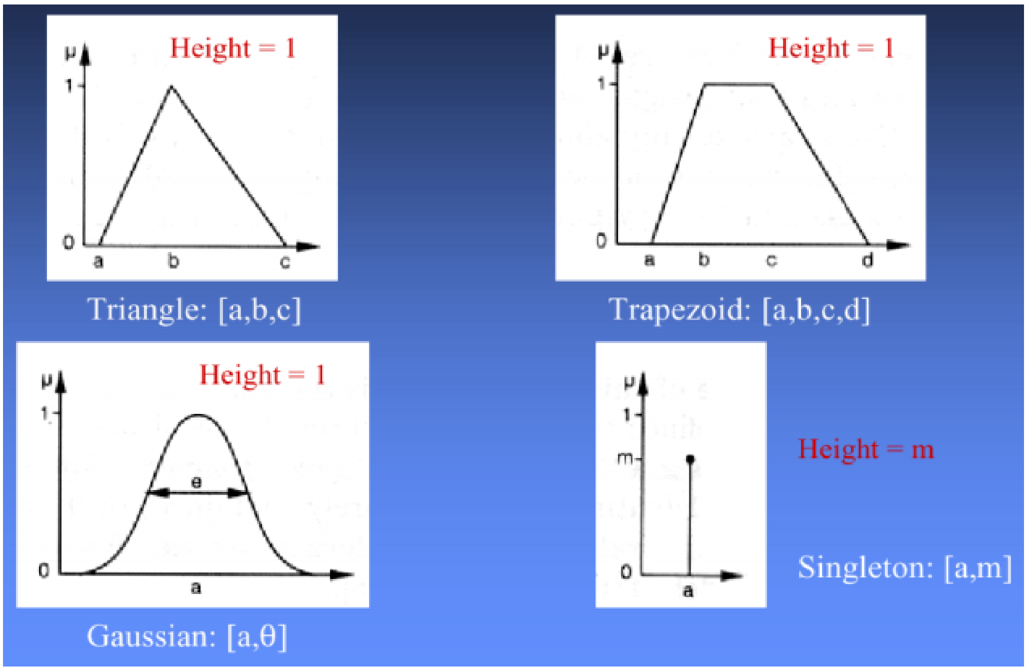


Fig 7

Normal Fuzzy Set

A normal fuzzy set is one that consists of at-least one element ‘x’ of universe whose membership value is unity. For fuzzy sets where only one element which has a membership value of unity, that particular element is called prototype of thefuzzy set or prototypical element.

Convex Fuzzy Set

Convex fuzzy set is described by a membership function whose membership values are strictly Monotonically Increasing or Monotonically Decreasing or Initially Monotonically Increasing then Monotonically Decreasing with the increase in the values of the elements of that particular fuzzy set.

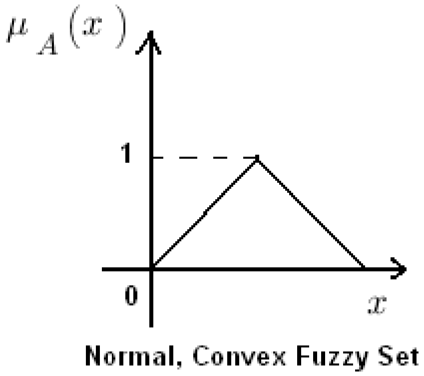


Fig 8

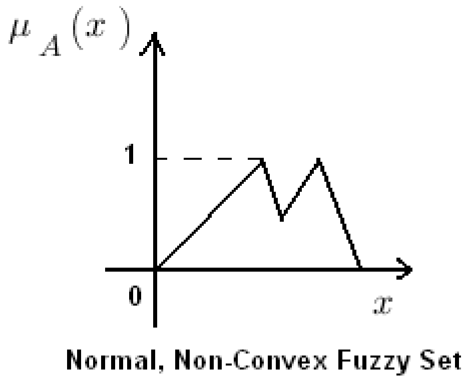


Fig 9

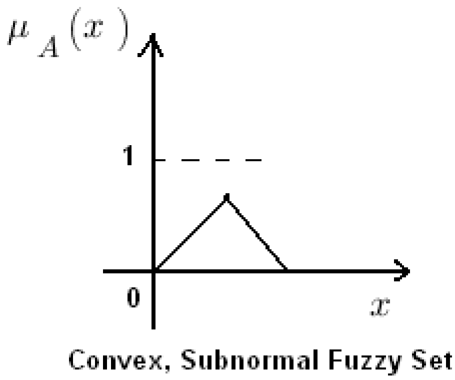
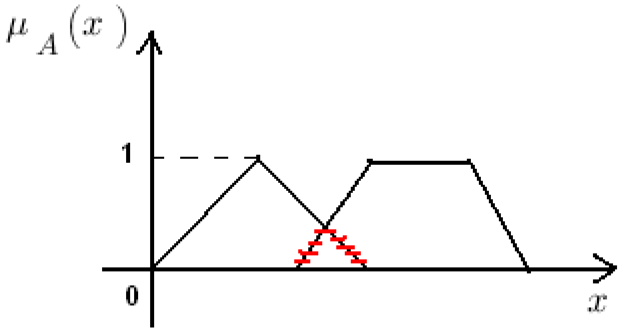


Fig 10

Note: The intersection of two convex fuzzy sets is also a convex fuzzy set.

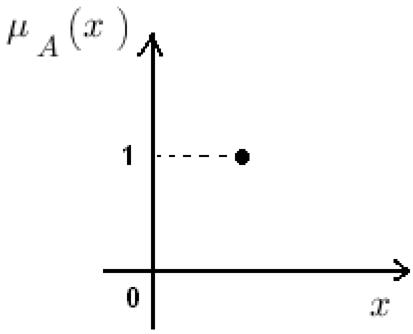


Intersection of Two Convex Fuzzy Set

Fig 11

Fuzzy Number

If 'A' is a convex single point normal fuzzy set defined on real line, then 'A' is called Fuzzy Number.



Convex Single Point Fuzzy Set

Fig 12

CHAPTER 4

SOFTWARE DESIGN

DATA FLOW

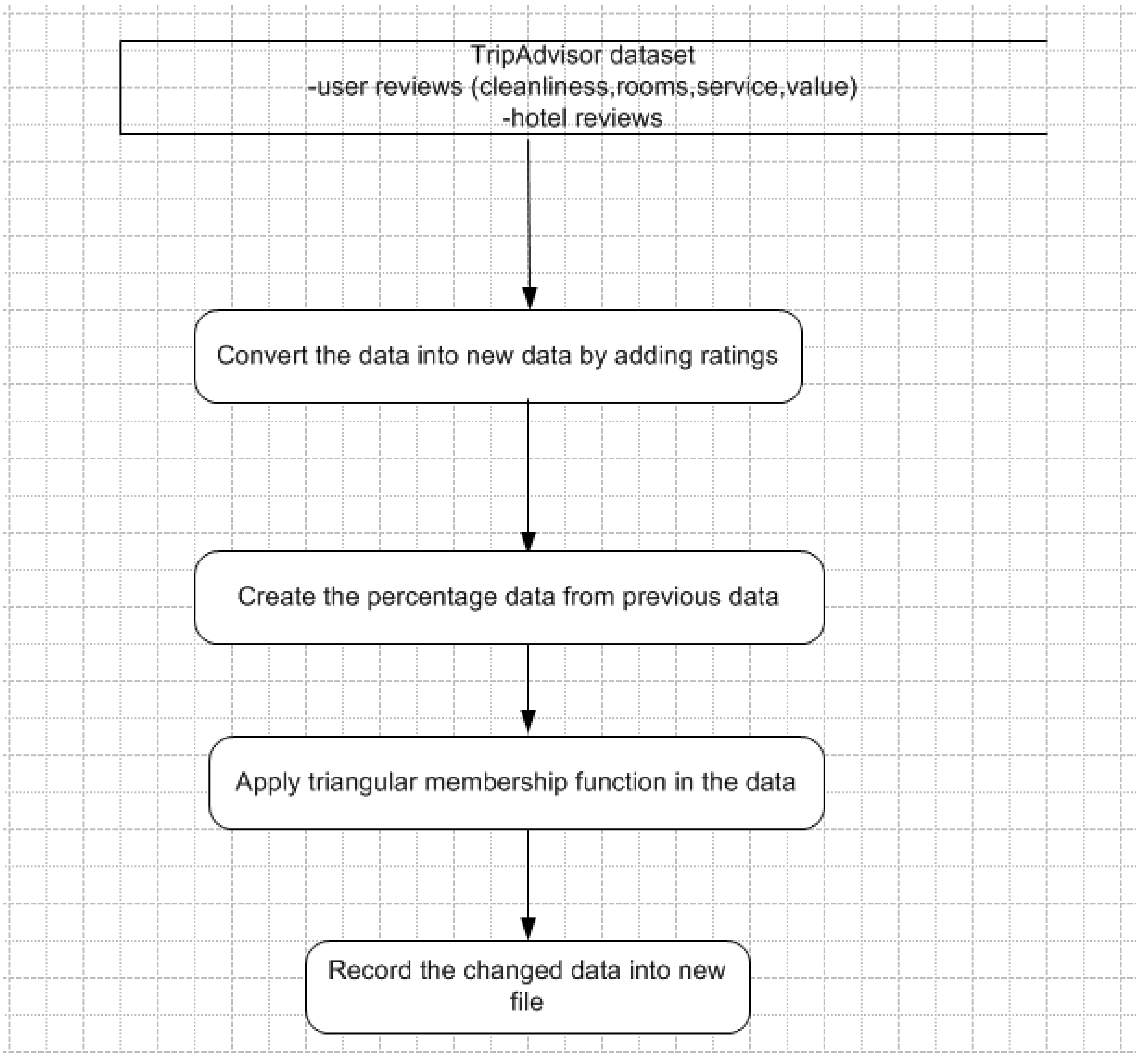


Fig 13

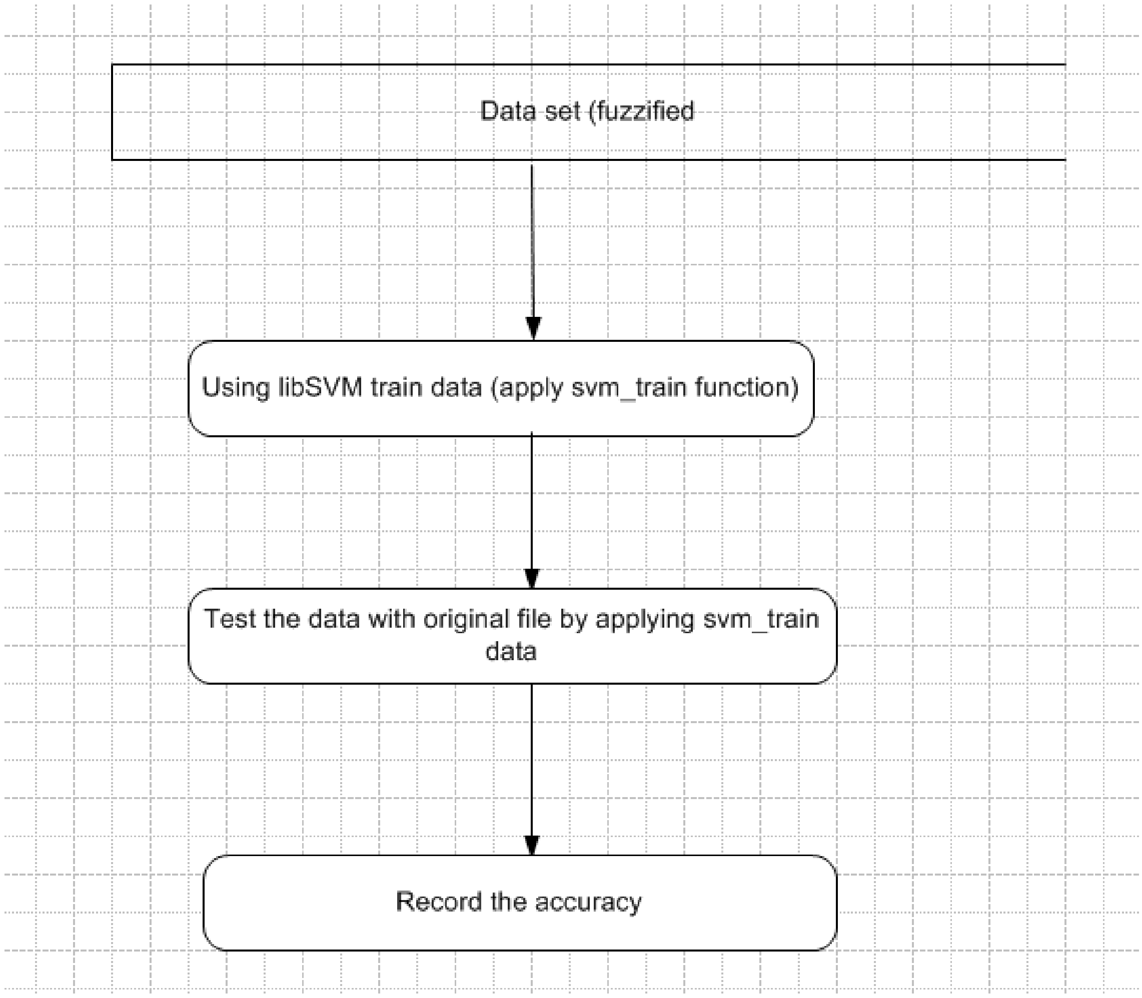

Data set (fuzzified
Using libSVM train data (apply svm_train function)
Test the data with original file by applying svm_train data
Record the accuracy

Fig 14

CHAPTER 5

SOFTWARE AND HARDWARE REQUIRMENTS

5.1 libSVM

It is an open source learning library for support vector machine written in C++ with C API. It implements the SMO algorithm for kernelized support vector. It is easy to use software for svm classification and regression.Two steps are

followed for accessing the libSVM software:-

1- Train a data set for creating a model.

2-Use the model for predicting the information about testing data set.

The sub-routines of LIBSVM contains svm train and svm predict. Svm train developes a two-class problems by decoupling a multi-class problem and one several times calls SVM train.

CHAPTER 6

CODING /CODE TEMPLATES

6.1 Original table

We took the original dataset table from TripAdvisor dataset in JSON format and converted it to CSV.

A sample of data from the original table is given in Fig1.

Ratings - Service	Ratings - Cleanliness	Ratings - Overall	Ratings - Value	Ratings - Sleep Quality	Ratings - Rooms	Ratings - Location	AuthorLocation	Title	Author	ReviewID	Content	Date
4	5	5	4	4	5	5	Boston	\u201cExcellent Hotel & Location\u201d	gowharr3	UR126946	We enjoy	29-Mar-12
5	5	5	5	5	5	5	Madison, Wisconsin	\u201cGreat Visit to Seattle!\u201d	Nancy W	UR126795	Great visit	27-Mar-12
5	5	5	5	4	4	5	Ketchikan, Alaska	\u201cExcellent in Everyway\u201d	Janet H	UR126715	Great Loca	27-Mar-12
3	4	5	3	3	3	5	Florida	\u201cGreat hotel, location & price.\u201	TimothyFl	UR126585	Accommo	24-Mar-12
1	1	4	1	4	1	1	Armstrong, BC	\u201cCool place\u201d	KarenArm	UR126067	Very cool	13-Mar-12
5	5	1	5	5	4	5		\u201cVery bad customer service!\u201d	Shane333	UR125893	I purchase	10-Mar-12
4	4	5	5	4	4	4		\u201cAn amazing historic hotel\u201d	Bnkruzn	UR125344	We stayec	28-Feb-12
5	5	5	4	5	5	5	Saskatchewan, Cana	\u201cSuper hotel when using Amtrak\u2	Teacherbe	UR125099	said: This	23-Feb-12
5	4	4	4	5	4	4	Kingston, Canada	\u201cGood price for a good location\u20	CandyGnc	UR124738	We went	17-Feb-12
5	5	5	5	5	2	5	Boise, Idaho, USA	\u201cJust passing thru\u201d	idahosanc	UR124685	Great littl	16-Feb-12
5	5	4	5	4	5	5	Ellicott City, MD	\u201cA conveniently-located downtown	CW2S	UR124359	We stayec	9-Feb-12
4	5	2	4	4	3	3	Duncan, Oklahoma	\u201cGOOD LODGING IN RIGHT PLACE\u2	jimmy62_	UR124231	Stayed tw	6-Feb-12
5	5	5	5	2	5	5	Boulder	\u201cHistoric gem of a hotel\u201d	BoulderIll	UR123915	The Best \	31-Jan-12
5	5	4	5	4	5	5	Modesto, California	\u201cvery comfortable\u201d	funloving	UR123521	We loved	23-Jan-12
5	5	4	2	5	2	5	Oak Park, CA	\u201cGreat location, nice room, friendly	suntravel	UR123449	My husba	22-Jan-12
4	5	5	5	5	4	5	Lisbon, Portugal	\u201cGreat experience!\u201d	rosariodu	UR123184	My husba	16-Jan-12
5	4	2	5	3	4	4	Longview, Washingt	\u201cterrible, terrible, terrible\u201d	Jody R	UR122179	If I hadn't	28-Dec-11
5	5	4	5	5	5	5	Vancouver, BC	\u201cOld but great!\u201d	Tasha M	UR121958	The staff	21-Dec-11
4	4	5	2	2	2	4	Nanaimo, Canada	\u201cExcellent hotel and location\u201d	Roy C	UR121926	Having sta	20-Dec-11
5	5	5	5	5	5	5	Vancouver, Canada	\u201cExcellent boutique hotel!\u201d	MikeGB2	UR121769	I have sta	15-Dec-11
5	5	3	5	4	5	4	Bellingham, Washin	\u201cDont pay for thier parking\u201d	Jennie S	UR121442	Some frie	6-Dec-11
4	4	5	4	4	3	5	Palm Harbor, Florida	\u201cGreat Hotel, Great Service and Staf	mcdonoth	UR121212	One of th	29-Nov-11
5	5	5	5	5	5	5	Kansas	\u201cA diamond in the rough\u201d	SWanjiru	UR121068	Don't judg	27-Nov-11
4	4	3	5	5	4	4	olympia, wa	\u201cgreat location for seahawks games\	trish0	UR120710	stayed at	17-Nov-11
5	5	4	5	3	5	4		\u201cGreat Find\u201d	txlnstr	UR120028	We were	1-Nov-11
5	5	4	5	5	5	5	Gainesville, GA	\u201cquick visit to seattle\u201d	mydogisfa	UR119966	Very close	31-Oct-11
5	5	5	5	5	5	5	Oakville, Canada	\u201cThey should all be like this!\u201d	DaddyHov	UR119772	I liked this	26-Oct-11
5	5	5	5	4	4	5	Vancouver, Canada	\u201clovely hotel close to the piers and (	BCisBeaut	UR119762	I've stayec	26-Oct-11
5	5	5	5	5	5	5	Boston, Massachuse	\u201cGreat Stay\u201d	wildorchi	UR119438	We stayec	18-Oct-11

Table 1

The original table contained following attributes

Reviews - Ratings - Sleep Quality

In this field the quality of sleep which a user gets is rated.

Reviews - Ratings - Service

In this field the quality of services which is provided to the user is rated.

Reviews – Author Location

In this field the author's / user location is noted.

Reviews – Author

The name of author/ user is written here in this field.

Reviews Hotel name

This field contains the hotel names that are rated by the user/ author.

Reviews Value

All the values in the fields are integer values which depicts the rating given by the author/ users.

Reviews Rooms

In this field the types of rooms that are provided to the user is rated.

Reviews – Cleanliness

In this field the cleanliness of the hotels and rooms that are provided to the user is rated.

Title

The title of the review that the user gave is cited here.

Review Id

The user's unique id which is given to the user when he/she creates an account on the Trip Advisor site.

Content The additional comments that a user wrote about a hotel is under this column.

Date

An easy column to understand. The values here state the date on which the review was given by the user.

6.2 Summed Table

Hotel Name	Sum - Ratings - Service	Sum - Ratings - Cleanliness	Sum - Ratings - Overall	Sum - Ratings - Value	Sum - Ratings - Sleep Quality	Sum - Ratings - Rooms	Sum - Ratings - Location
BEST WESTERN PLUS Pioneer Square H	991	1012	1021	948	462	845	886
Grace Inn Phoenix	117	122	123	111	40	92	110
BEST WESTERN PLUS Eagle Rock Inn	47	54	44	47	22	39	42
Comfort Inn Near Old Town Pasadena -	624	609	851	607	418	540	464
Rodeway Inn & Suites Pacific Coast Hig	861	869	855	846	698	638	594
Dunes Inn - Sunset	30	36	41	37	29	35	37
Hollywood Hotel	902	1034	1013	987	579	901	780
BEST WESTERN PLUS Hollywood Hills H	457	520	503	488	234	457	434
BEST WESTERN Hollywood Plaza Inn	320	349	330	347	173	283	378
Days Inn Hollywood/Universal Studios	391	393	394	418	242	322	406
BEST WESTERN PLUS InnSuites Phoeni	227	228	238	229	118	207	193
Econo Lodge Hollywood	437	453	610	430	27	429	321
Hollywood Orchid Suites	1563	1527	1642	1670	656	1309	1585
Hollywood Roosevelt Hotel - A Thomp	1786	1900	1962	1604	884	1540	1781
Comfort Inn Near the Sunset Strip	274	268	280	272	126	231	267
Travelodge Hollywood-Vermont/Suns	295	328	322	354	188	266	308
BEST WESTERN Terrace Inn	202	198	211	208	118	160	185
BEST WESTERN PLUS Boston - The Inn a	555	547	587	518	286	460	549
Sea Bay Hotel	599	571	596	594	391	438	484
W Los Angeles - Westwood	1183	1247	1268	1062	787	1096	1073
A Victory Inn & Suites Phoenix North	75	68	70	66	21	66	78
BEST WESTERN PLUS Royal Palace Inn &	168	173	185	168	85	136	131
Kyoto Grand Hotel and Gardens	950	1031	977	938	650	793	868
Dunes Inn - Wilshire	36	43	38	44	31	31	45
BEST WESTERN PLUS Dragon Gate Inn	155	181	170	166	61	146	129
BEST WESTERN PLUS Hotel & Conferen	256	240	263	246	151	179	164
Howard Johnson Los Angeles	81	74	80	80	56	72	81
The Historic Mayfair Hotel	193	199	201	206	88	164	138

Table 2

In this field the reviews of the quality of sleep which a user gets is summed.ta

Sum - Reviews - Ratings - Service

In this field the reviews of quality of services which is provided to the user is summed.

Sum - Reviews – Author Location

In this field the reviews of author's / user location is summed.

Sum - Reviews – Author

The name of author/ user is written here in this field.

Sum - Reviews - Hotel name

This field contains the summed value of hotel names that are rated by the user/ author.

Sum - Reviews – Cleanliness

In this field the reviews for cleanliness of the hotels and rooms that are provided to the user is summed together.

Sum - Reviews Value

All the values in the fields are integer values which depicts the rating given by the author/ users. And those values are summed together.

Sum - Reviews Rooms

In this field the reviews of types of rooms that are provided to the user is rated and summed together.

6.3 Percentage data table

In this field the percentage value of all the attributes is calculated and then stored in a new file.

Hotel Name	%- Ratings - Service	% - Ratings - Cleanliness	% - Ratings - Overall	% - Ratings - Value	% - Ratings - Sleep Quality	% - Ratings - Rooms	% - Ratings - Location
BEST WESTERN PLUS	91.75925926	92.8440367	87.63948498	86.57534247	88.84615385	85.35353535	89.49494949
Grace Inn Phoenix	68.82352941	64.21052632	55.90909091	58.42105263	61.53846154	51.11111111	75.86206897
BEST WESTERN PLUS	72.30769231	83.07692308	67.69230769	72.30769231	73.33333333	70.90909091	76.36363636
Comfort Inn Near Ol	47.63358779	46.13636364	64.22641509	46.51340996	78.86792453	42.1875	36.39215686
Rodeway Inn & Suite	93.58695652	94.45652174	89.52879581	92.45901639	89.48717949	86.21621622	80.27027027
Dunes Inn - Sunset	60	72	63.07692308	67.27272727	72.5	70	74
Hollywood Hotel	63.97163121	73.33333333	63.71069182	69.75265018	77.71812081	67.7443609	65.82278481
BEST WESTERN PLUS	74.30894309	82.53968254	74.51851852	78.08	79.3220339	78.79310345	78.1981982
BEST WESTERN Holly	71.11111111	76.7032967	69.47368421	75.43478261	69.2	68.19277108	88.94117647
Days Inn Hollywood/	69.82142857	72.77777778	69.12280702	74.64285714	75.625	65.71428571	88.26086957
BEST WESTERN PLUS	87.30769231	86.03773585	79.33333333	86.41509434	84.28571429	81.17647059	85.77777778
Econo Lodge Hollyw	45.7591623	47.43455497	62.56410256	45.02617801	54	45.15789474	34.51612903
Hollywood Orchid Su	86.11570248	83.21525886	82.72040302	90.02695418	78.56287425	75.66473988	96.94189602
Hollywood Roosevel	71.87122736	76.30522088	67.77202073	64.41767068	72.45901639	68.90380313	89.49748744
Comfort Inn Near th	74.05405405	72.43243243	69.13580247	73.51351351	66.31578947	68.95522388	83.4375
Travelodge Hollywo	68.60465116	72.88888889	67.08333333	78.66666667	68.36363636	67.34177215	77
BEST WESTERN Terra	77.69230769	74.71698113	71.52542373	77.03703704	81.37931034	66.66666667	77.08333333
BEST WESTERN PLUS	84.09090909	82.87878788	81.52777778	77.31343284	79.44444444	77.96610169	91.5
Sea Bay Hotel	85.57142857	82.15827338	80.54054054	85.4676259	79.79591837	76.84210526	87.20720721
W Los Angeles - Wes	80.75085324	85.11945392	78.03076923	72.49146758	85.08108108	82.09737828	85.15873016
A Victory Inn & Suite	55.55555556	50.37037037	43.75	48.88888889	38.18181818	50.76923077	65
BEST WESTERN PLUS	78.13953488	80.46511628	68.51851852	78.13953488	77.27272727	71.57894737	74.85714286
Kyoto Grand Hotel ai	74.80314961	79.61389961	72.10332103	72.15384615	79.26829268	70.17699115	76.47577093
Dunes Inn - Wilshire	45	57.33333333	47.5	58.66666667	62	44.28571429	69.23076923
BEST WESTERN PLUS	63.26530612	70.98039216	60.71428571	66.4	64.21052632	63.47826087	69.72972973
BEST WESTERN PLUS	76.41791045	75	71.08108108	73.43283582	83.88888889	68.84615385	64.31372549
Howard Johnson Los	55.86206897	51.03448276	44.44444444	55.17241379	62.22222222	51.42857143	64.8
The Historic Mayfair	55.94202899	55.27777778	47.29411765	56.43835616	56.77419355	51.25	52.0754717

Table 3

Percentage value - Reviews - Ratings - Sleep Quality

In this field the reviews of the quality of sleep which a user gets is summed and then its percentage value is calculated.

Percentage value - Reviews - Ratings - Service

In this field the reviews of quality of services which is provided to the user is summed and then its percentage value is calculated.

Percentage value - Reviews – Author Location

In this field the reviews of author's / user location is summed and then its percentage value is calculated.

Percentage value - Reviews – Author

The name of author/ user is written here in this field and the percentage value is calculated.

Percentage value - Reviews - Hotel name

This field contains the summed value of hotel names that are rated by the user/ author. And its percentage value is calculated.

Percentage value - Reviews – Cleanliness

In this field the reviews for cleanliness of the hotels and rooms that are provided to the user is summed together and then its percentage value is calculated.

Percentage value - Reviews Value

All the values in the fields are integer values which depicts the rating given by the author/ users. And those values are summed together. And then its percentage value is calculated.

Percentage value- Reviews Rooms

In this field the reviews of types of rooms that are provided to the user is rated and summed together and then its percentage value is calculated.

6.4 Fuzzy data table

After calculation the percentage value of all the attributes in the table we applied the triangular membership function for the fuzzification of the data and after fuzzifying the data we stored the data in the new file.

Hotel Name	Fuzzy-Ratings-Service	Fuzzy-Ratings-Cleanliness	Fuzzy-Ratings-Overall	Fuzzy-Ratings-Value	Fuzzy-Ratings-Sleep Quality	Fuzzy-Ratings-Rooms	Fuzzy-Ratings-Location
BEST WESTERN PLUS	0.659134045	0.691204032	0.695789722	0.720017739	0.806602339	0.696697092	0.802789744
Grace Inn Phoenix	0.859007161	0.702138805	0.641355229	0.68242436	0.636181814	0.575902268	0.999231081
BEST WESTERN PLUS	0.977474234	0.823197214	0.913210042	0.964823895	0.917569923	0.942691374	0.991322958
Comfort Inn Near Ol	0.494507789	0.453567821	0.811974708	0.523047837	0.943323026	0.473280058	0.350504371
Rodeway Inn & Suite	0.638081483	0.673378747	0.673706913	0.651816001	0.799161358	0.686005742	0.926584041
Dunes Inn - Sunset	0.657272086	0.919256247	0.783179923	0.882263706	0.889764774	0.964116178	0.918987968
Hollywood Hotel	0.734964291	0.970631845	0.798798205	0.961116137	0.962114981	0.966547964	0.67939509
BEST WESTERN PLUS	0.962257104	0.831935728	0.900860194	0.848155873	0.936101901	0.790370859	0.959830623
BEST WESTERN Holly	0.933271163	0.940381592	0.975736341	0.89791235	0.794432834	0.984552059	0.809280502
Days Inn Hollywood/	0.889897342	0.948543345	0.962752413	0.913964237	0.998318539	0.89264717	0.817399474
BEST WESTERN PLUS	0.716730152	0.778151439	0.812938403	0.722075512	0.863822847	0.75356207	0.848468347
Econo Lodge Hollyw	0.476617856	0.465401947	0.7709823	0.508223427	0.531924593	0.503122901	0.340012244
Hollywood Orchid Su	0.733902003	0.820976747	0.760710445	0.678377321	0.948236736	0.844516956	0.724633904
Hollywood Roosevel	0.960873848	0.94881742	0.915836205	0.806122973	0.888440719	0.991282189	0.802760236
Comfort Inn Near th	0.968764562	0.93531229	0.963227135	0.937874179	0.726409523	0.9899737	0.8799926
Travelodge Hollywo	0.852516346	0.952880251	0.893633633	0.837858735	0.773430587	0.950935557	0.980167598
BEST WESTERN Terra	0.883482091	0.98403709	0.965792944	0.867100793	0.904726176	0.925857122	0.978725341
BEST WESTERN PLUS	0.765037239	0.826398543	0.778317307	0.861998258	0.93417424	0.803997943	0.780134976
Sea Bay Hotel	0.742019492	0.838253022	0.793520623	0.734486601	0.928683265	0.823290219	0.830300789
W Los Angeles - Wes	0.822604874	0.791585682	0.834984779	0.960616977	0.853265923	0.740241609	0.856585301
A Victory Inn & Suite	0.587745761	0.494584585	0.490637505	0.548607614	0.395814477	0.571157527	0.662028225
BEST WESTERN PLUS	0.874024073	0.867495539	0.941183072	0.847099396	0.969597073	0.927504171	0.954262996
Kyoto Grand Hotel ai	0.949885622	0.882981351	0.952536765	0.968374125	0.936950707	0.95986897	0.989338863
Dunes Inn - Wilshire	0.469735214	0.58098752	0.528975692	0.686740575	0.643908718	0.493977115	0.762214784
BEST WESTERN PLUS	0.71983219	0.883495931	0.729972873	0.857505648	0.683679551	0.823311846	0.776065693
BEST WESTERN PLUS	0.911591629	0.977570874	0.976239188	0.939630216	0.869188206	0.992753292	0.648207692
Howard Johnson Los	0.592064982	0.501700882	0.497312197	0.630048345	0.647696416	0.580379234	0.657940054
The Historic Mayfair	0.593202196	0.552493719	0.526716072	0.649472918	0.566063037	0.577852413	0.472358134

Table 4

The attributes in the table are as followed.

Fuzzy value - Reviews - Ratings - Sleep Quality

In this field the reviews of the quality of sleep which a user gets is summed and then its percentage value is calculated and then after applying triangular membership function in the values a new data is created.

Fuzzy value - Reviews - Ratings - Service

In this field the reviews of quality of services which is provided to the user is summed and then its percentage value is calculated and then after applying triangular membership function in the values a new data is created.

Fuzzy value - Reviews – Author Location

In this field the reviews of author's / user location is summed and then its percentage value is calculated and then after applying triangular membership function in the values a new data is created.

Fuzzy value - Reviews – Author

The name of author/ user is written here in this field and the percentage value is calculated and then after applying triangular membership function in the values a new data is created.

Fuzzy value - Reviews - Hotel name

This field contains the summed value of hotel names that are rated by the user/ author. And its percentage value is calculated and then after applying triangular membership function in the values a new data is created.

Fuzzy value - Reviews – Cleanliness

In this field the reviews for cleanliness of the hotels and rooms that are provided to the user is summed together and then its percentage value is calculated and then after applying triangular membership function in the values a new data is created.

Fuzzy value - Reviews Value

All the values in the fields are integer values which depicts the rating given by the author/ users. And those values are summed together. And then its percentage value is calculated and then after applying triangular membership function in the values a new data is created.

Fuzzy value- Reviews Rooms

In this field the reviews of types of rooms that are provided to the user is rated and summed together and then its percentage value is calculated and then after applying triangular membership function in the values a new data is created.

CHAPTER 7

TESTING DATA SETS

We tested parts of the data set to get an idea of the accuracy achieved as we increased the no of data classes for libSVM to train.

DATA SET A

The data set A contained 25 hotel values converted into libSVM class form as depicted in fig15.

```
+1 1:0.659134045 2:0.69120403 3:0.6957897 4:0.720017739 5:0.8066023
+2 1:0.69669709 2:0.80278974 3:0.859007161 4:0.70213881 5:0.6413552
+3 1:0.68242436 2:0.6361818 3:0.57590227 4:0.99923108 5:0.977474234
+4 1:0.82319721 2:0.91321 3:0.964823895 4:0.9175699 5:0.94269137
+5 1:0.99132296 2:0.494507789 3:0.45356782 4:0.8119747 5:0.523047837
+6 1:0.943323 2:0.47328006 3:0.35050437 4:0.638081483 5:0.67337875
+7 1:0.6737069 2:0.651816001 3:0.7991614 4:0.68600574 5:0.92658404
+8 1:0.657272086 2:0.91925625 3:0.7831799 4:0.882263706 5:0.8897648
+9 1:0.96411618 2:0.91898797 3:0.734964291 4:0.97063185 5:0.7987982
+10 1:0.961116137 2:0.962115 3:0.96654796 4:0.67939509 5:0.962257104
+11 1:0.83193573 2:0.9008602 3:0.848155873 4:0.9361019 5:0.79037086
+12 1:0.95983062 2:0.933271163 3:0.94038159 4:0.9757363 5:0.89791235
+13 1:0.7944328 2:0.98455206 3:0.8092805 4:0.889897342 5:0.94854334
+14 1:0.9627524 2:0.913964237 3:0.9983185 4:0.89264717 5:0.81739947
+15 1:0.716730152 2:0.77815144 3:0.8129384 4:0.722075512 5:0.8638228
+16 1:0.75356207 2:0.84846835 3:0.476617856 4:0.46540195 5:0.7709823
+17 1:0.508223427 2:0.5319246 3:0.5031229 4:0.34001224 5:0.733902003
+18 1:0.82097675 2:0.7607104 3:0.678377321 4:0.9482367 5:0.84451696
+19 1:0.7246339 2:0.960873848 3:0.94881742 4:0.9158362 5:0.806122973
+20 1:0.8884407 2:0.99128219 3:0.80276024 4:0.968764562 5:0.93531229
+21 1:0.9632271 2:0.937874179 3:0.7264095 4:0.9899737 5:0.8799926
+22 1:0.852516346 2:0.95288025 3:0.8936336 4:0.837858735 5:0.7734306
+23 1:0.95093556 2:0.9801676 3:0.883482091 4:0.98403709 5:0.9657929
+24 1:0.867100793 2:0.9047262 3:0.92585712 4:0.97872534 5:0.765037239
+25 1:0.82639854 2:0.7783173 3:0.861998258 4:0.9341742 5:0.80399794
```

Fig15. Shows testing data form for libSVM

The recommendation accuracy for this set was a high 95% due to the low no of values to be tested.

DATA SET B

In the data set B we increased the no of hotel values to 300. A sample of the data is given in Fig16.

```
+271 1:0.80734633 2:0.666645973 3:0.73795812 4:0.9323547 5:0.656949366
+272 1:0.5646584 2:0.83671484 3:0.42512629 4:0.60221474 5:0.62778199
+273 1:0.9725673 2:0.552243362 3:0.8014851 4:0.62476665 5:0.38840872
+274 1:0.77132525 2:0.71331539 3:0.9026927 4:0.904014429 5:0.9731802
+275 1:0.79625333 2:0.59294702 3:0.640559157 4:0.6816593 5:0.6587583
+276 1:0.661984898 2:0.82508 3:0.63977042 4:0.73214683 5:0.778096025
+277 1:0.73639722 2:0.7540419 3:0.894942497 4:0.8405445 5:0.72084693
+278 1:0.7996031 2:0.880090017 3:0.80642906 4:0.7619717 5:0.872712053
+279 1:0.7734188 2:0.78766173 3:0.58906345 4:0.751502401 5:0.72380574
+280 1:0.7332864 2:0.864965272 3:0.8583318 4:0.69485851 5:0.7651156
+281 1:0.66683294 2:0.66966633 3:0.6791393 4:0.802584044 5:0.7962233
+282 1:0.64412636 2:0.7724532 3:0.696600859 4:0.81772232 5:0.8242431
+283 1:0.812112333 2:0.7896677 3:0.84773714 4:0.52252036 5:0.525856193
+284 1:0.56247866 2:0.8935628 3:0.532105034 4:0.8594156 5:0.58456972
+285 1:0.38663677 2:0.642240679 3:0.77596099 4:0.7374673 5:0.698189838
+286 1:0.8166392 2:0.80227745 3:0.77617962 4:0.690977791 5:0.74858769
+287 1:0.7509694 2:0.788882753 3:0.8823856 4:0.78660742 5:0.74132638
+288 1:0.460966823 2:0.42614178 3:0.7162052 4:0.397111803 5:0.7999328
+289 1:0.44810689 2:0.29422413 3:0.805458416 4:0.73999211 5:0.8064585
+290 1:0.982565008 2:0.8336697 3:0.73523249 4:0.81181585 5:0.999205542
+291 1:0.99405654 2:0.95488 3:0.815361035 4:0.9967187 5:0.84972951
+292 1:0.97734928 2:0.844660214 3:0.90329296 4:0.9095078 5:0.921392625
+293 1:0.8566024 2:0.81140747 3:0.97109912 4:0.933356902 5:0.8416336
+294 1:0.9102474 2:0.954822332 3:0.9062427 4:0.80074566 5:0.83707865
+295 1:0.779477842 2:0.7147961 3:0.942559 4:0.818841251 5:0.9916334
+296 1:0.69484779 2:0.54396809 3:0.514248345 4:0.48896631 5:0.9728966
+297 1:0.48748003 2:0.7572473 3:0.49036803 4:0.27768532 5:0.959537812
+298 1:0.98638246 2:0.8532447 3:0.825955662 4:0.9324314 5:0.92996199
+299 1:0.92168502 2:0.741811651 3:0.7525966 4:0.7178064 5:0.854639755
+300 1:0.81703 2:0.76567332 3:0.9100549 4:0.701184323 5:0.67035934
```

Fig 16. Depicts a part of the data set B

The accuracy of the data set was around 89%.

DATA SET C

For data set C we used almost half of the hotels. This data set contains 5000 values. A sample of data is given in Fig 17

```
+4974 1:0.76599338 2:0.7660498 3:0.758766685 4:0.8616952 5:0.77129294
+4975 1:0.88817604 2:0.874515474 3:0.83125907 4:0.8115623 5:0.993854373
+4976 1:0.8156177 2:0.89984177 3:0.89330446 4:0.766497234 5:0.78610637
+4977 1:0.9326735 2:0.815361035 3:0.832972 4:0.86573698 5:0.99606286
+4978 1:0.793136948 2:0.81429061 3:0.8807364 4:0.761743987 5:0.9991317
+4979 1:0.88555856 2:0.92313419 3:0.840410553 4:0.81061268 5:0.8651527
+4980 1:0.931010408 2:0.9463228 3:0.83245274 4:0.91115236 5:0.790966673
+4981 1:0.78549454 2:0.8365981 3:0.763048828 4:0.8960257 5:0.77129294
+4782 1:0.85869165 2:0.872057182 3:0.85620384 4:0.8547052 5:0.945547177
+4983 1:0.9891534 2:0.7016114 3:0.81109589 4:0.984971844 5:0.96780116
+4984 1:0.9026927 2:0.877503124 3:0.9827525 4:0.93487441 5:0.75300634
+4985 1:0.910394315 2:0.85369017 3:0.8910199 4:0.911282887 5:0.9175699
+4986 1:0.81918771 2:0.87722306 3:0.591036281 4:0.50253519 5:0.4946206
+4987 1:0.567731191 2:0.3598313 3:0.53828753 4:0.39221228 5:0.76816061
+4988 1:0.75709141 2:0.7614341 3:0.804160136 4:0.8863192 5:0.76449741
+4989 1:0.8814503 2:0.93421078 3:0.96662308 4:0.9060059 5:0.968030797
+4990 1:0.9154311 2:0.97952834 3:0.85962948 4:0.778816471 5:0.7647485
+4991 1:0.7831656 2:0.88234821 3:0.8405836 4:0.71810033 5:0.79750105
+4992 1:0.680647697 2:0.45457963 3:0.4930197 4:0.56527945 5:0.6932751
+4993 1:0.50555605 2:0.60112873 3:0.668216849 4:0.63590067 5:0.8646588
+4994 1:0.61743461 2:0.8478087 3:0.69567774 4:0.49241597 5:0.654090764
+4995 1:0.7061786 2:0.6830304 3:0.685295316 4:0.7849561 5:0.67560425
+4996 1:0.75780368 2:0.792899385 3:0.82319721 4:0.8642147 5:0.801679112
+4997 1:0.9675938 2:0.88481658 3:0.93078934 4:0.998344232 5:0.92564684
+4998 1:0.8492874 2:0.883381793 3:0.7593682 4:0.79395388 5:0.85084198
+4999 1:0.740935763 2:0.69012413 3:0.667087 4:0.813207599 5:0.8156177
+5000 1:0.57590227 2:0.90194022 3:0.519106783 4:0.35521297 5:0.4823146
```

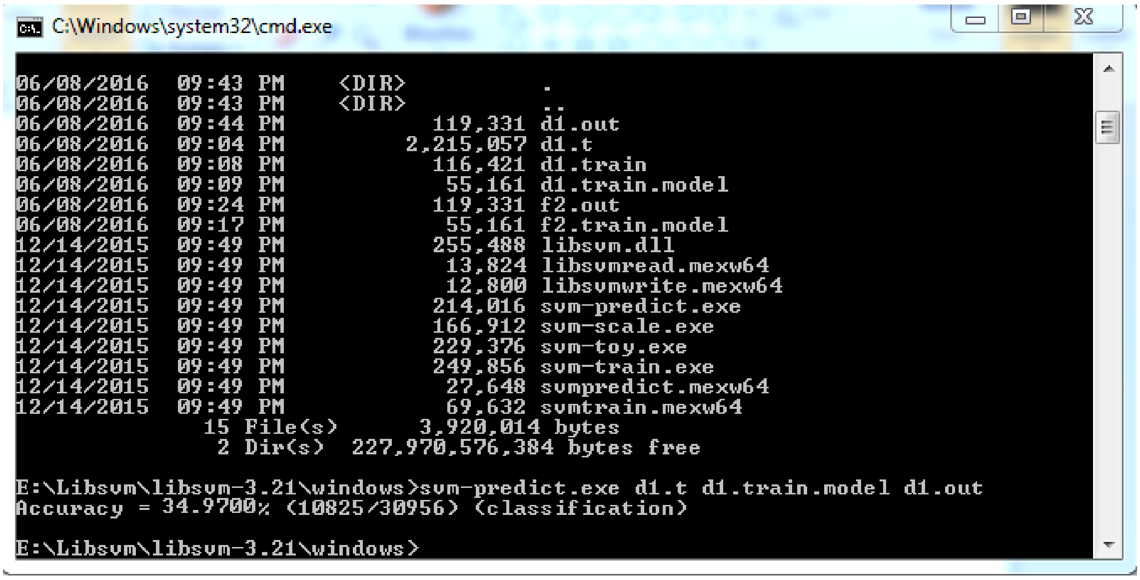
Fig17. Contains sample data from data set C

The accuracy dropped as expected for data set C due to the tremendous increase in testing values but it was still around 75% which is an amazing number.

CHAPTER 8

OUTPUT SCREEN

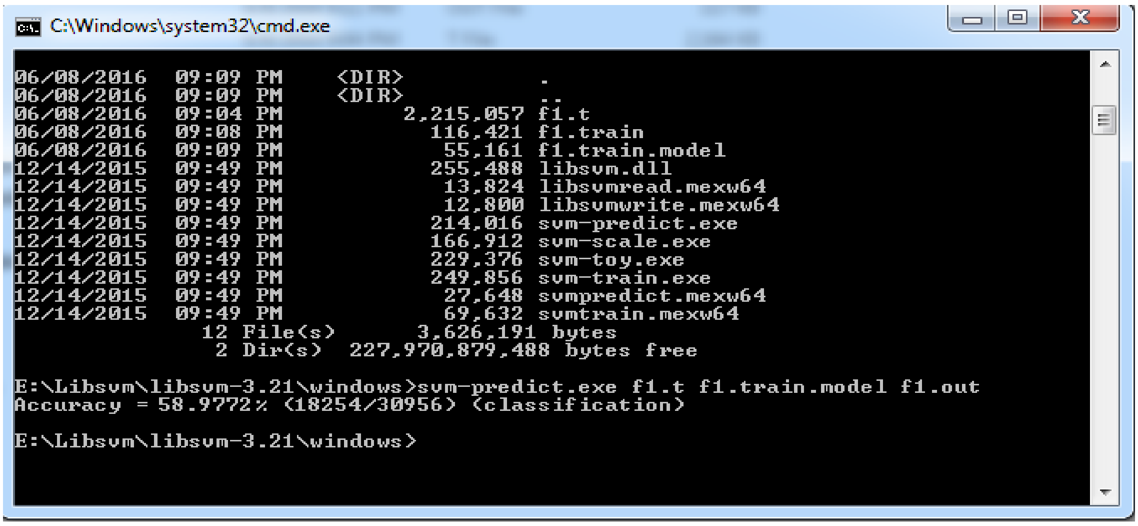
Sleep Quality



```
C:\Windows\system32\cmd.exe

06/08/2016  09:43 PM    <DIR>          .
06/08/2016  09:43 PM    <DIR>          ..
06/08/2016  09:44 PM           119,331 d1.out
06/08/2016  09:04 PM         2,215,057 d1.t
06/08/2016  09:08 PM           116,421 d1.train
06/08/2016  09:09 PM            55,161 d1.train.model
06/08/2016  09:24 PM           119,331 f2.out
06/08/2016  09:17 PM            55,161 f2.train.model
12/14/2015  09:49 PM           255,488 libsvm.dll
12/14/2015  09:49 PM            13,824 libsvmread.mexw64
12/14/2015  09:49 PM            12,800 libsvmwrite.mexw64
12/14/2015  09:49 PM           214,016 svm-predict.exe
12/14/2015  09:49 PM           166,912 svm-scale.exe
12/14/2015  09:49 PM           229,376 svm-toy.exe
12/14/2015  09:49 PM           249,856 svm-train.exe
12/14/2015  09:49 PM            27,648 svmpredict.mexw64
12/14/2015  09:49 PM            69,632 svmtrain.mexw64
              15 File(s)      3,920,014 bytes
               2 Dir(s)  227,970,576,384 bytes free

E:\Libsvm\libsvm-3.21\windows>svm-predict.exe d1.t d1.train.model d1.out
Accuracy = 34.9700% (10825/30956) (classification)

E:\Libsvm\libsvm-3.21\windows>
```

Fig 18

1. Output screen for %data accuracy



```
C:\Windows\system32\cmd.exe

06/08/2016  09:09 PM    <DIR>          .
06/08/2016  09:09 PM    <DIR>          ..
06/08/2016  09:04 PM         2,215,057 f1.t
06/08/2016  09:08 PM           116,421 f1.train
06/08/2016  09:09 PM            55,161 f1.train.model
12/14/2015  09:49 PM           255,488 libsvm.dll
12/14/2015  09:49 PM            13,824 libsvmread.mexw64
12/14/2015  09:49 PM            12,800 libsvmwrite.mexw64
12/14/2015  09:49 PM           214,016 svm-predict.exe
12/14/2015  09:49 PM           166,912 svm-scale.exe
12/14/2015  09:49 PM           229,376 svm-toy.exe
12/14/2015  09:49 PM           249,856 svm-train.exe
12/14/2015  09:49 PM            27,648 svmpredict.mexw64
12/14/2015  09:49 PM            69,632 svmtrain.mexw64
              12 File(s)      3,626,191 bytes
               2 Dir(s)  227,970,879,488 bytes free

E:\Libsvm\libsvm-3.21\windows>svm-predict.exe f1.t f1.train.model f1.out
Accuracy = 58.9772% (18254/30956) (classification)

E:\Libsvm\libsvm-3.21\windows>
```

Fig 19

2.Output screen for fuzzy data accuracy

Service

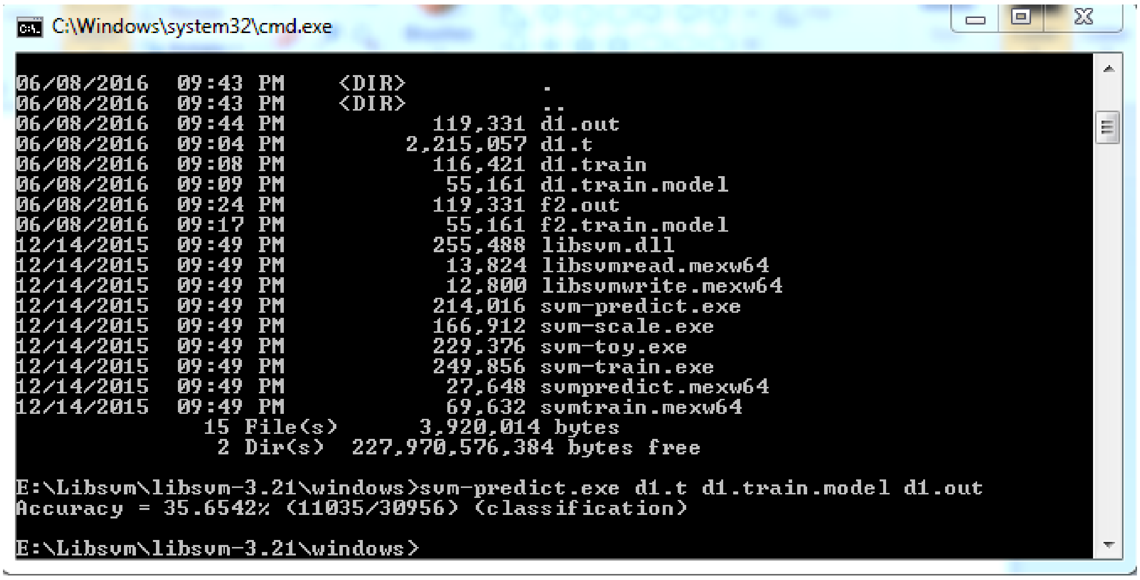


Fig 20

1.Output screen for %data accuracy

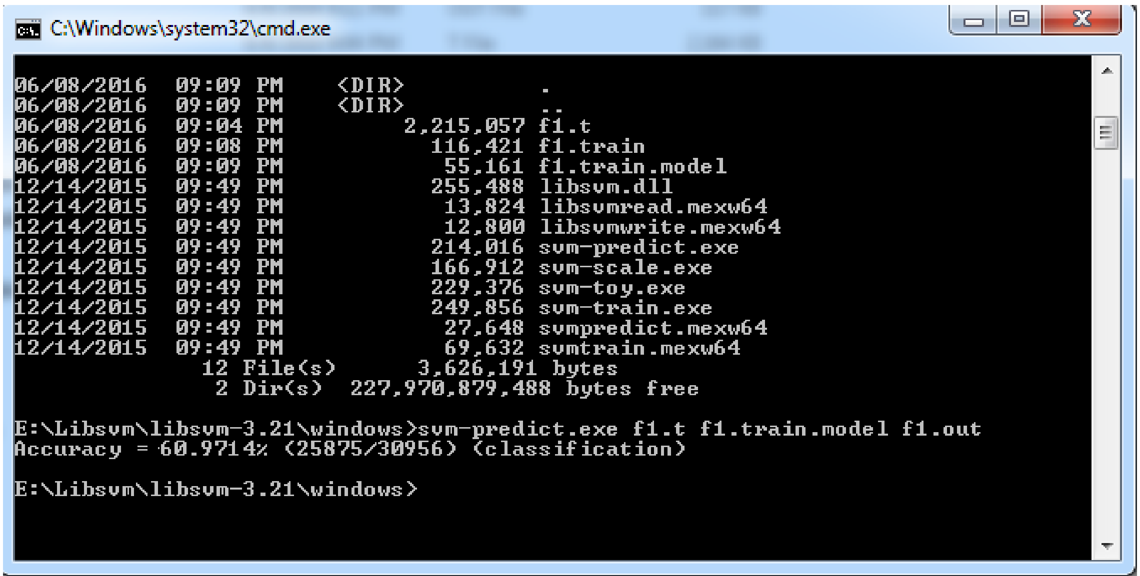


Fig 21

2. Output screen for fuzzy data accuracy

Value

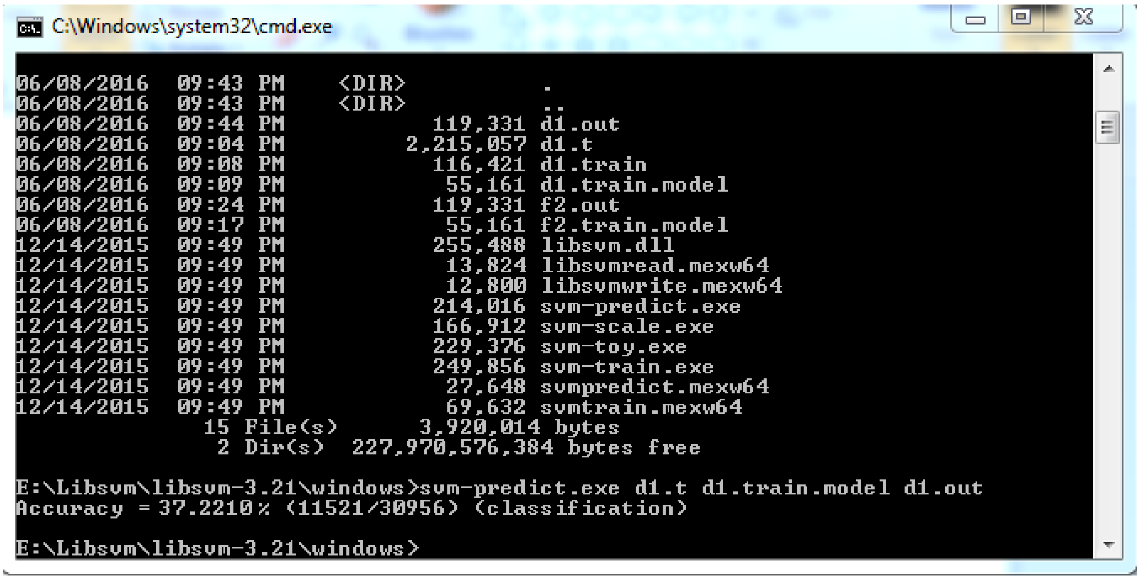


Fig 22

1.Output Screen for % data accuracy

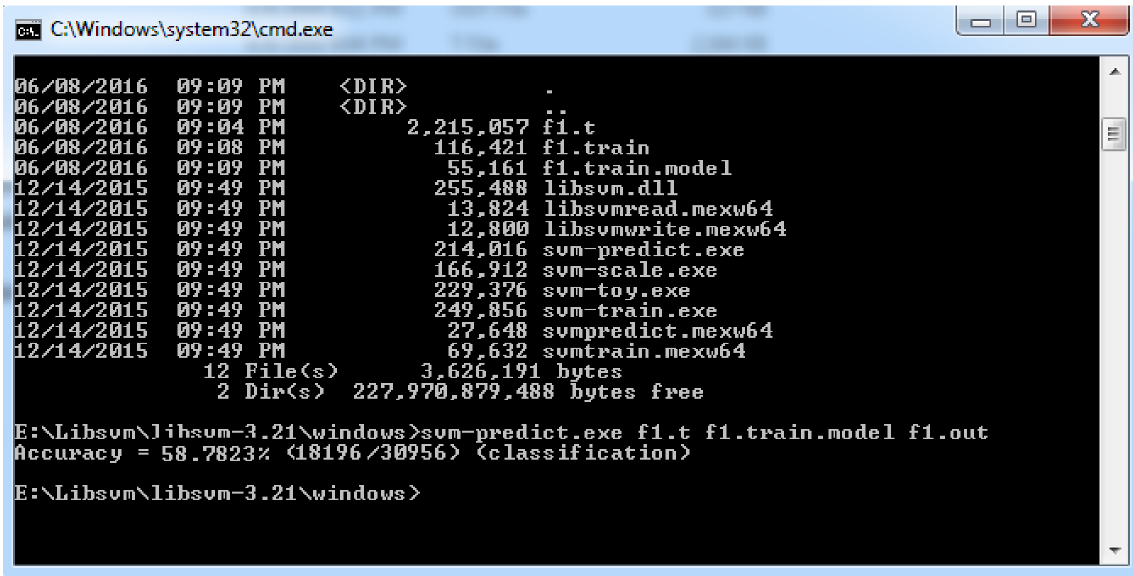


Fig 23

2.Output Screen for fuzzy data accuracy

Rooms

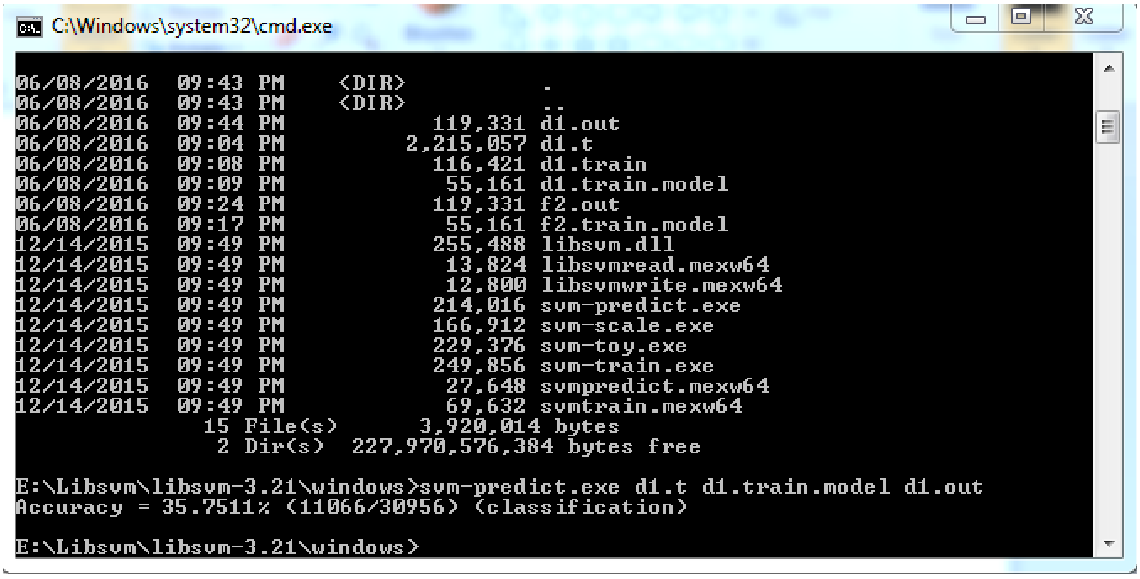


Fig 24

1.Output Screen for %data accuracy

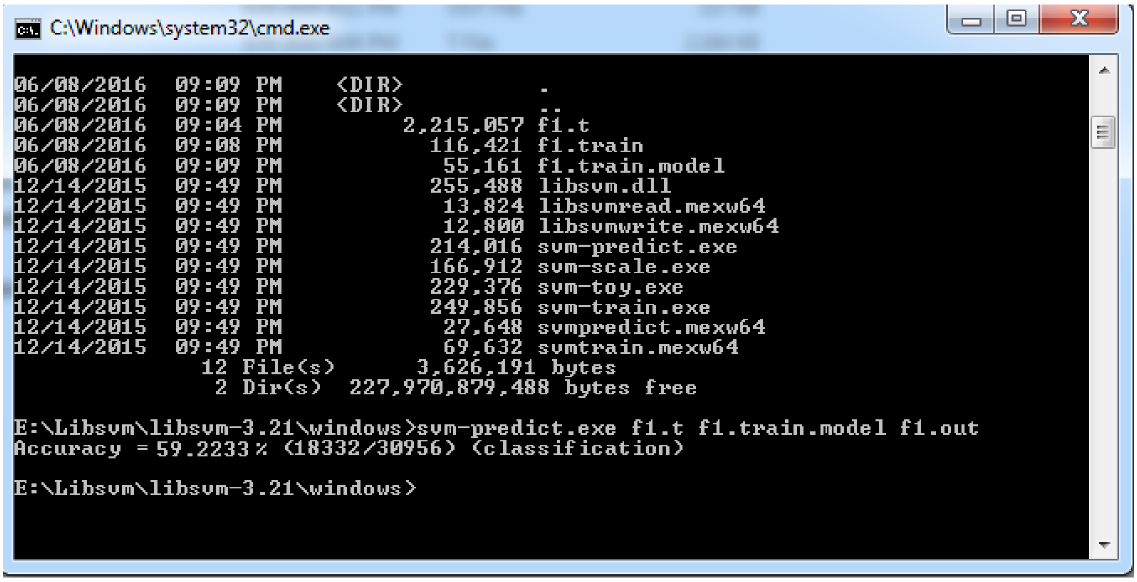


Fig 25

2.Output Screen for fuzzy data accuracy

Cleanliness

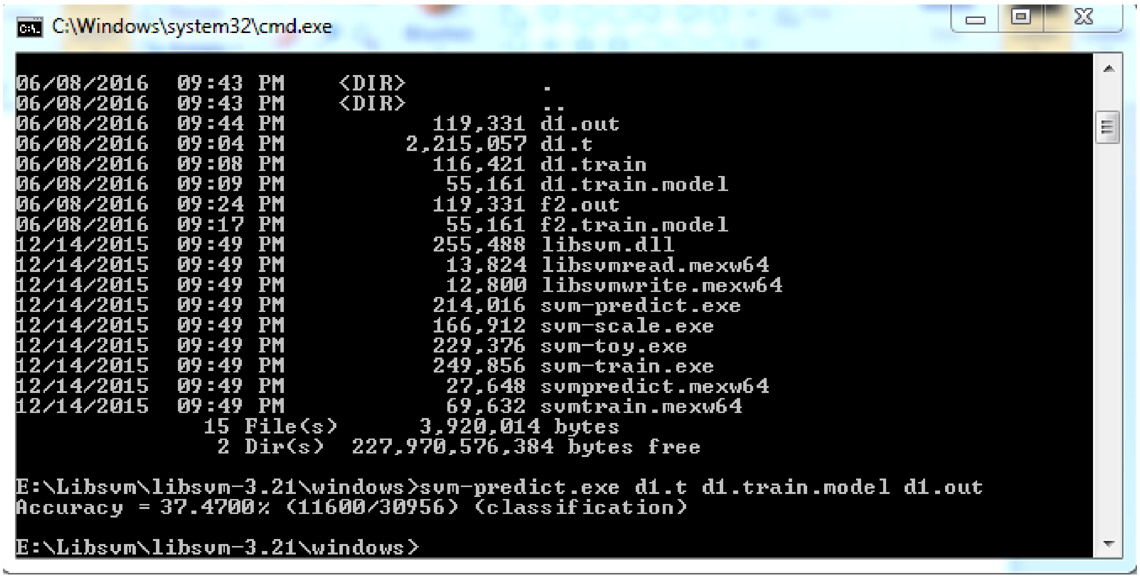


Fig 26

1.Output screen for %data

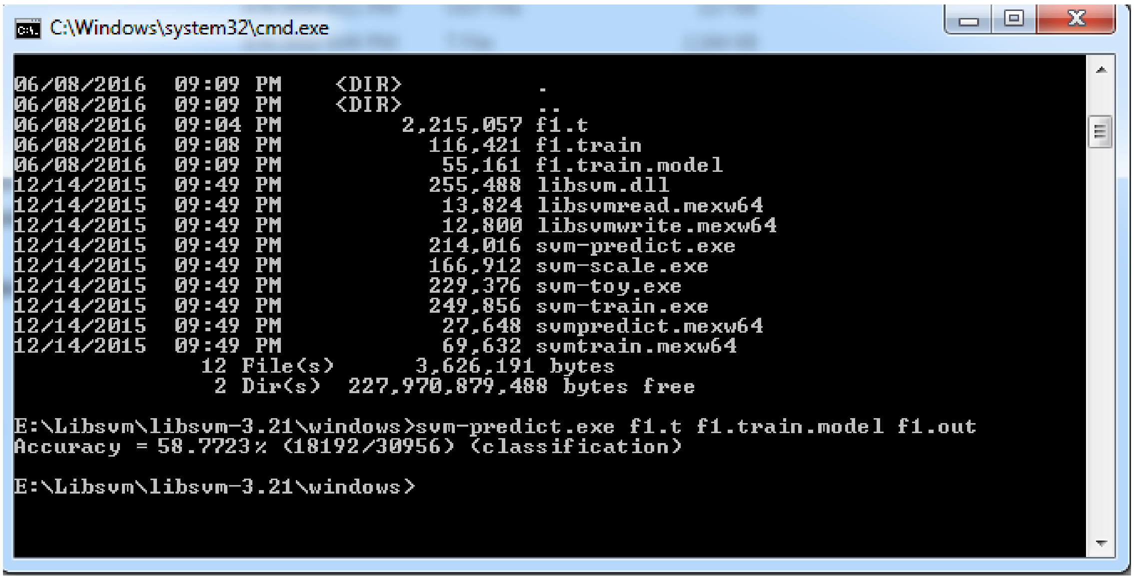


Fig 27

2.Output screen for fuzzy data

Location

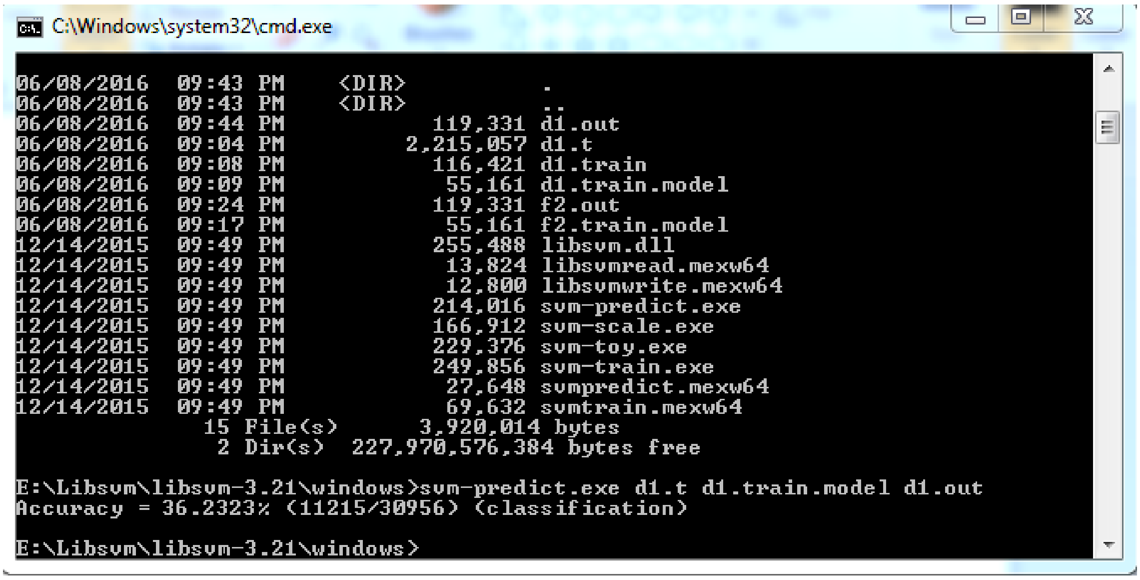


Fig 28

1.Output Screen for %data

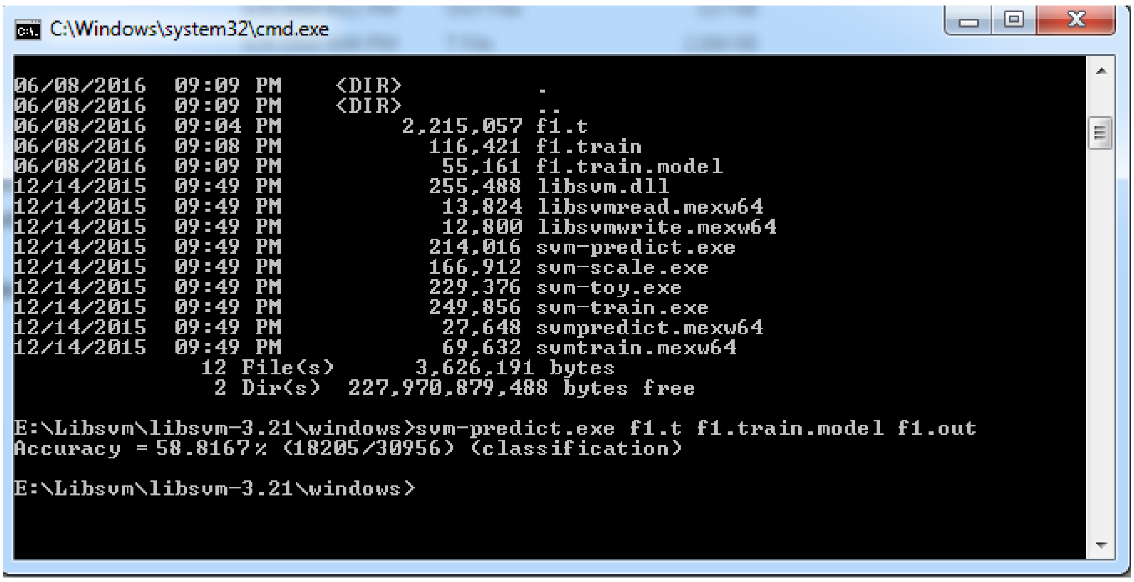
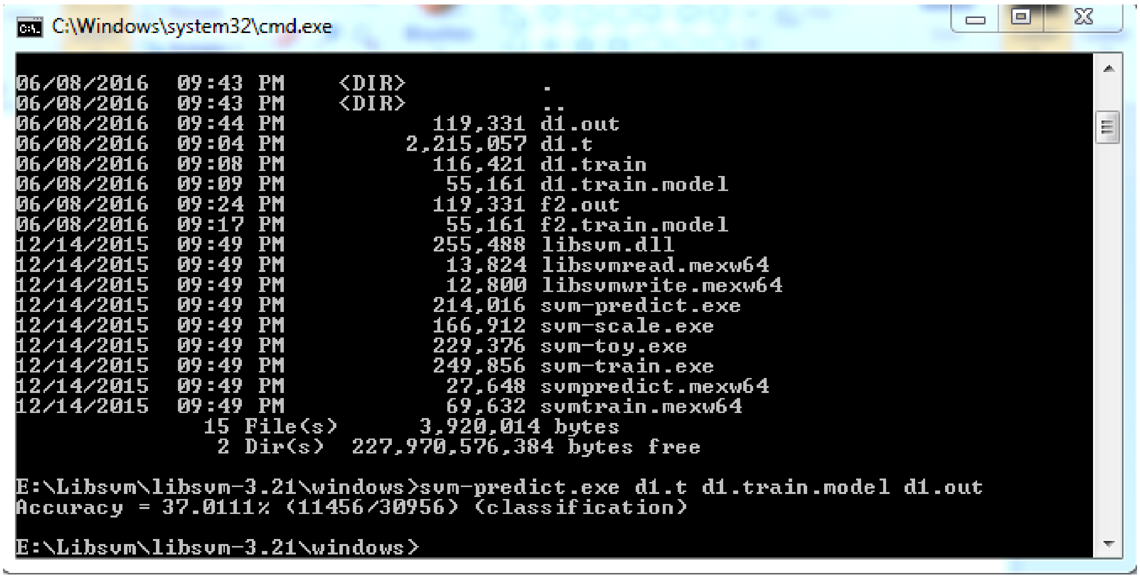


Fig 29

2.Output Screen for fuzzy data accuracy

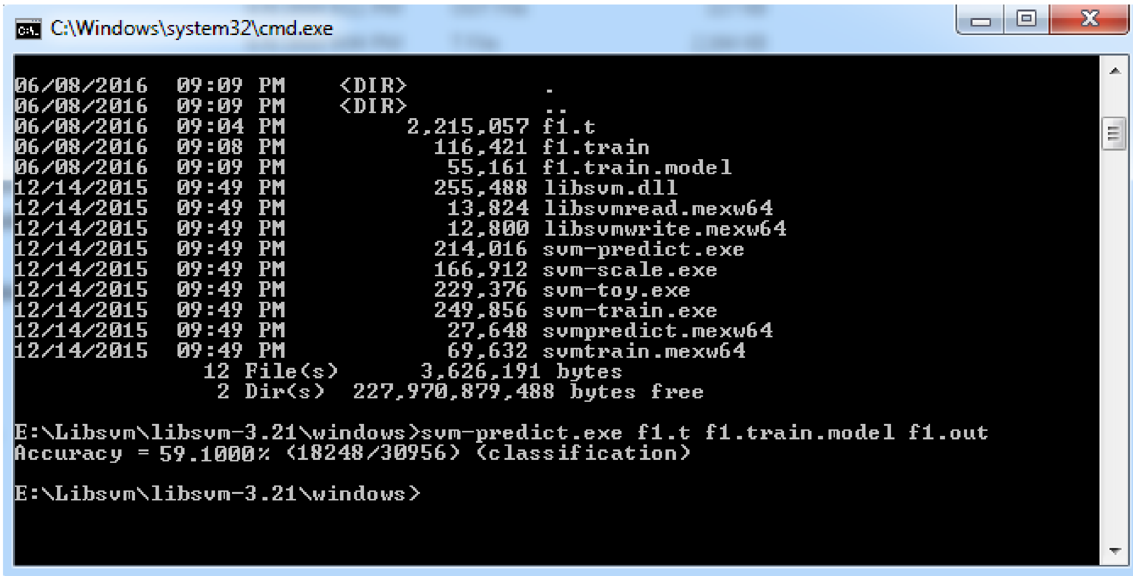
OVERALL



```
C:\Windows\system32\cmd.exe
06/08/2016  09:43 PM    <DIR>          .
06/08/2016  09:43 PM    <DIR>          ..
06/08/2016  09:44 PM           119,331 d1.out
06/08/2016  09:04 PM         2,215,057 d1.t
06/08/2016  09:08 PM           116,421 d1.train
06/08/2016  09:09 PM            55,161 d1.train.model
06/08/2016  09:24 PM           119,331 f2.out
06/08/2016  09:17 PM            55,161 f2.train.model
12/14/2015  09:49 PM           255,488 libsvm.dll
12/14/2015  09:49 PM            13,824 libsvmread.mexw64
12/14/2015  09:49 PM            12,800 libsvmwrite.mexw64
12/14/2015  09:49 PM           214,016 svm-predict.exe
12/14/2015  09:49 PM           166,912 svm-scale.exe
12/14/2015  09:49 PM           229,376 svm-toy.exe
12/14/2015  09:49 PM           249,856 svm-train.exe
12/14/2015  09:49 PM            27,648 svmpredict.mexw64
12/14/2015  09:49 PM            69,632 svmtrain.mexw64
              15 File(s)      3,920,014 bytes
               2 Dir(s)  227,970,576,384 bytes free

E:\Libsvm\libsvm-3.21\windows>svm-predict.exe d1.t d1.train.model d1.out
Accuracy = 37.0111% (11456/30956) (classification)

E:\Libsvm\libsvm-3.21\windows>
```

Fig 30

1.Output Screen for %data



```
C:\Windows\system32\cmd.exe
06/08/2016  09:09 PM    <DIR>          .
06/08/2016  09:09 PM    <DIR>          ..
06/08/2016  09:04 PM         2,215,057 f1.t
06/08/2016  09:08 PM           116,421 f1.train
06/08/2016  09:09 PM            55,161 f1.train.model
12/14/2015  09:49 PM           255,488 libsvm.dll
12/14/2015  09:49 PM            13,824 libsvmread.mexw64
12/14/2015  09:49 PM            12,800 libsvmwrite.mexw64
12/14/2015  09:49 PM           214,016 svm-predict.exe
12/14/2015  09:49 PM           166,912 svm-scale.exe
12/14/2015  09:49 PM           229,376 svm-toy.exe
12/14/2015  09:49 PM           249,856 svm-train.exe
12/14/2015  09:49 PM            27,648 svmpredict.mexw64
12/14/2015  09:49 PM            69,632 svmtrain.mexw64
              12 File(s)      3,626,191 bytes
               2 Dir(s)  227,970,879,488 bytes free

E:\Libsvm\libsvm-3.21\windows>svm-predict.exe f1.t f1.train.model f1.out
Accuracy = 59.1000% (18248/30956) (classification)

E:\Libsvm\libsvm-3.21\windows>
```

Fig 31

2.Output Screen for fuzzy data accuracy

Table 5

Rating Name	% Data Accuracy	Fuzzy Data Accuracy
Service	35.65	60.97
Cleanliness	37.47	58.77
Sleep Quality	34.97	58.97
Location	36.23	58.81
Rooms	35.75	59.22
Overall	37.01	59.1
Value	37.22	58.78

CHAPTER 9

CONCLUSION

In this project, we studied how fuzzy data analysis leads to better recommendations. We proposed an efficient form of data which leads to better recommendations for the user. Theoretical analysis is provided to guarantee the efficiency and effectiveness of our method. Extensive experimental results on real-world data sets as in this case done on the Trip Advisor data sets also confirm our theoretical findings.

CHAPTER 10

FUTURE SCOPE

This concept of recommendation technique of using fuzzy data can be used in many recommendation systems, not only in the trip advisor dataset but also in several other data sets.

For eg. Several ecommerce websites can use fuzzy data form in their datasets to enhance their recommendations. As our findings have shown this form can increase the accuracy of recommendations by almost twice of the original accuracy.

Hence, using this proposed method can bring about more efficient recommendation systems as it proposes a method of enhancing recommendations.

References/Bibliography

[1] C. Aggarwal, and P. Yu, A Framework for Clustering

Massive Text and Categorical Data Streams, SDM, (2006).

[2] C. Aggarwal, Data Streams: Models and Algorithms,

Springer, (2007).

[3] C. Aggarwal, Social Network Data Analytics, Springer,

(2011).

[4] J. Allan, R. Papka, and V. Lavrenko, On-line new event

detection and tracking, SIGIR Conf., (1998).

Document Collections, SIGIR Conf., (1992).

[5] https://www.csie.ntu.edu.tw/~cjlin/libsvm/

[6] A fuzzy logic approach to analyzing gene expression data, PETER
J. WOOLF, YIXIN WANG

[7] Multiscale event detection in social media,Xiaowen Dong1 · Dimitrios Mavroeidis2 · Francesco Calabrese3 · Pascal Frossard4

[8] FUZZY DATA MINING AND GENETIC ALGORITHMS APPLIED TO INTRUSION DETECTION,Susan M. Bridges Bridges@cs.msstate.edu Rayford B. Vaughn vaughn@cs.msstate.edu 23rd National Information Systems Security Conference October 16-19, 2000

[9] Pairwise Preference Regression for Cold-start Recommendation,Seung-Taek Park,Samsung Advanced Institute of Technology,Mt. 14-1, Nongseo-dong, Giheung-gu,Yongin-si, Gyunggi-do 446-712, South Korea,park.seungtaek@gmail.com,Wei Chu,Yahoo! Labs,4401 Great America, Parkway, Santa Clara, CA 95054, USA,chuwei@yahoo-inc.com

[10] Advances in Social Media Analysis,edited by Mohamed Medhat Gaber, Mihaela Cocea, Nirmalie Wiratunga, Ayse Goker

[11]http://times.cs.uiuc.edu/ - tripadvisor dataset in JSON format.

[12] Multiscale event detection in social media,Xiaowen Dong1 · Dimitrios Mavroeidis2 · Francesco Calabrese3 · Pascal Frossard4

[13] FUZZY DATA MINING AND GENETIC ALGORITHMS APPLIED TO INTRUSION DETECTION,Susan M. Bridges Bridges@cs.msstate.edu Rayford B. Vaughn vaughn@cs.msstate.edu 23rd National Information Systems Security Conference October 16-19, 2000

[14] Pairwise Preference Regression for Cold-start Recommendation,Seung-TaekPark,Samsung Advanced Institute of Technology,Mt. 14-1, Nongseo-dong, Giheung-gu,Yongin-si, Gyunggi-do 446-712, South Korea,park.seungtaek@gmail.com,WeiChu,Yahoo! Labs,4401 Great America, Parkway, Santa Clara, CA 95054, USA,chuwei@yahoo-inc.com

[15] Proceedings of the ACM RecSys'09 Workshop on
Recommender Systems & the SocialWeb edited by,DietmarJannach, Werner Geyer, Jill Freyne, Sarabjot Singh Anand,
Casey Dugan, BamshadMobasher, Alfred Kobsa,October 25, 2009,New York, NY, USA