9-18-2018

Accessible POI Recommendation Using Adaptive Aggregation of Binary Ratings

Bidur Subedi
University of Windsor

Accessible POI Recommendation Using Adaptive Aggregation of Binary Ratings

By

Bidur Subedi

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for the
Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2018

Accessible POI Recommendation Using Adaptive Aggregation of Binary Ratings

by

Bidur Subedi

APPROVED BY:

G. Bhandari
Odette School of Business

D. Wu
School of Computer Science

X. Yuan, Co-Advisor
School of Computer Science

S. Mavromoustakos, Co-Advisor
School of Computer Science

September 10, 2018

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Everyone needs one or more forms of accessibility at some point in life due to age, medical conditions, accidents, etc. People with accessibility needs have the right to accessible services, as well as the right to information about accessibility at various places or Points of Interest (POI). While most popular POI recommendation services do not take accessibility into account, some of them only consider a few specific needs, such as ramp for wheelchair users.

However, different users have different accessibility needs regarding the structure of the building, special aid devices, and facilities to be able to independently visit a place. The proposed system focuses on finding the personalized accessibility score for a (user, POI) pair. It can be used with other factors such as historical behavior, social influence, geographical conditions, etc. to recommend accessible places. It uses time decaying aggregate on the crowd-sourced binary rating data to find accurate approximation of current accessibility status for each accessibility criteria. Also, we propose a tunnel-based algorithm to detect the trend of binary stream data to update the rate of decay. This ensures that the calculated aggregate adapts to change in the accessibility status of the place.

DEDICATION

To my beloved family:

Parents: Keshab and Bidhya

Sister: Binita

for their love and support.

ACKNOWLEDGEMENT

I want to express my gratitude to my advisors Dr. Stephanos Mavromoustakos and Dr. Xiaobu Yuan for their support, encouragement and advice. I am grateful for their time and immense knowledge without which my Master's would be incomplete. I would also like to express my gratitude to my committee members Dr. Gokul Bhandari and Dr. Dan Wu for their advice and support on my thesis.

My sincere thank goes to all faculties and staffs at the School of Computer Science for their assistance, motivation and support.

Finally, I would like to thank my family and friends for supporting me over the course of my academic career.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1:
INTRODUCTION

1.1 Overview

Recommendation Systems have been helping people is a number of domains such as e-commerce, social networks, music and videos, news, information retrieval, etc. With the data available from Location-Based Social Networks (LBSNs), smartphones, and the crowdsourced data contributed by users, recommendation systems can help people discover attractive and interesting places. (Xie, et al., 2016). Applications such as Google Maps, FourSquare, Facebook, etc. have been helping people find interesting places by tracking their preference and various features of places. Such Point of Interest (POI) recommendation systems use factors, such as distance and geographical factors (Ye, Yin, Lee, & Lee, 2011), activities of related people (Chen, Li, Cheung, & Li, 2016), current location and movement pattern (Cheng, Yang, Lyu, & King, 2013).

In addition to interest, social and geographic factors, people with disability have additional needs and preferences when they visit any place. Accessibility aids, such as wheelchair ramp, accessible entrance, accessible toilet, elevators for multi-storied buildings enable a person with disability access the facilities of places independently (Imrie, 2005). Though some POI recommendation systems consider one or more accessibility factors to recommend places (AXS Map, 2015) (Mobasheri, Deister, & Dieterich, 2017), there is still a huge potential of accessible POI recommendation systems to help people with multiple disabilities find places they could enjoy independently.

This research explores techniques to analyze the crowdsourced ratings to determine the confidence that the place meets various accessibility needs of users. The ratings are considered to be stream data, and we will use the damped/time-fading window model to compute the confidence of fulfillment of accessibility criteria by a place. This model gives higher emphasis to recent information allowing us to find relevant confidence in the status of the place. Similarly, we enhance the algorithm proposed by (Santos, Almeida, Martins, Gonçalves, & José, 2017) to use confidence calculated from crowdsourced ratings to calculate a personalized accessibility score for a user and a place.

1.2 Motivation

Access to information is one of the most important accessibility tools for people with disabilities. Ability to know the accessibility status of places, and find places that cater accessibility needs not only enables disabled people to live a dignified life, but also generates awareness about accessibility in a public space (Mobasheri, Deister, & Dieterich, 2017). While some existing applications have been collecting and using data to help people find accessible places for some specific accessibility criteria (Access Now, 2017) space (Mobasheri, Deister, & Dieterich, 2017), we realized there is a gap in the services available for general people and people with disabilities. An example of wheel chair based recommendation is illustrated in figure 1:

Figure 1: Wheelmap: Online tool that helps find wheelchair accessible places. (Mobasheri, Deister, & Dieterich, 2017)

One of the important recent works in this area is "Using POI functionality and accessibility levels for delivering personalized tourism recommendations" by (Santos, Almeida, Martins, Gonçalves, & José, 2017). The algorithm proposed in the work for POI recommendations assumes the accessibility score of each POI would be defined in the system by an expert human user. But, this approach could be error prone as the output depends on the judgement of a single user. In addition, the process is not scalable as it requires an expert visit to each place before it is added to the system. Also, places are changing continuously and with more awareness in people, the accessibility of places are improving. If an expert user needs to visit the place to update the accessibility status of places, the recommendation system could be producing outdated results until then.

To solve the problem, we would investigate techniques to model the accessibility status of various factors of Point of Interests (POIs) using explicit feedback given by users. The value of status can be used to recommend personalized places based on the disability profile of users. This ensures that people with disabilities have access to information about private and public places before they visit.

1.3 Scope of Thesis

In this work, we analyze techniques to model the accessibility status of a place based on the feedback of other users. The model could be used in conjunction with the preference of the user to recommend accessible places. The key contribution of this work would be:

i. Improve the POI recommendation algorithm proposed by (Santos, Almeida, Martins, Gonçalves, & José, 2017) by automating the calculation of accessibility confidence of places based on crowdsourced ratings.
ii. Propose tunnel-based adaptive aggregation technique for binary ratings based on the algorithm used by (Gorawski, Gorawska, & Pasterak, 2017).
iii. Design an architecture of a recommendation system to recommend accessible POIs for people with multiple disabilities.

We will generate a dataset to cover various scenarios for places, ratings and person profiles and analyze the performance and results of the system when that data is fed. The test data will compose of different combinations of user profiles and ratings across a duration of time which are both consistent as well as random. We will also analyze the result of the system when some noise is introduced in the rating.

The major limitations of this work are:

i. We do not have access to real movement data of people with disabilities at this point. The project is aimed to develop a crowdsourced recommendation system which would help us collect data for future collection and analysis of real data.
ii. With lack of a real disability profile, we cannot analyze the relationship between accessibility ratings and the satisfaction of the person from the service provided by the place. The rating prediction could be improved by using the correlation between the user's profile, accessibility profile of place as well as the check-in history of the user.

1.4 Structure of Thesis

The remaining of the thesis is organized as follows: First part of Chapter 2 discusses the disability and government policies surrounding disabilities and accessibility. We have also reviewed basic concepts of recommendation systems and common recommendation algorithms from literature. Next, the thesis talks about Point of Interest (POI) recommendation as one of the application areas of recommendation systems. The next part of Chapter 2 provides the current status of POI recommendation for people with disabilities and discusses the work by (Santos, Almeida, Martins, Gonçalves, & José, 2017) on accessible POI recommendation. This is followed by brief overview of rating aggregation techniques. Chapter 3 discusses the implementation details of our approach. It is classified into Knowledge layer, Rating Aggregation Layer (pre-processing), and the recommendation layer. Chapter 4 discusses the data used for experiments and results. Chapter 5 provides the summary of the work with a conclusion and future directions.

CHAPTER 2:
REVIEW OF RELATED TOPICS

2.1 Disability and Accessibility

Disability is the physical or mental limitation or the gap between an individual's capabilities and the demand of the environment where s/he is living (Pope & Tarlov, 1991). Depending on the type of disability, people face multiple barriers in their everyday life that prevents them from performing daily activities without assistance. So, it is not the physical condition of the people, but the barriers that prevent them from performing their work independently is what makes them disabled. To make them independent, the products, services, physical infrastructures as well as the policy and attitude of people should cater to their need.

It is estimated that 14% of the total population of the world live with some form of disability (World Health Organization, 2011). They are facing physical, psychological and financial barriers that not only hinder their daily life but also restrict their access to education, health services, rehabilitation, employment and quality life. Three out of four Canadians with disabilities have reported more than one type of disability (Statistics Canada, 2012) which adds further barriers and sets them back on accessing services and facilities. Table 1 lists the proportion of people with disabilities, aged 15 or older living with co-occurring disabilities:

Table 1: Co-occurring disabilities, by type, aged 15 years or older with disabilities, Canada (Statistics Canada, 2012)

Type of Disability	Percent									
	Pain-related	**Flexibility**	**Mobility**	**Mental health-**	**Dexterity**	**Hearing**	**Seeing**	**Learning**	**Memory**	**Pain-Related**
Pain-related	...	64.9	61.3	30.2	30.7	22.1	21.1	17.3	18.9	2.9
Flexibility	83.7	...	72.4	31.8	37.7	24.6	24.3	19.5	21.6	3.8
Mobility	82.9	76.0	...	29.7	36.1	24.8	24.6	18.8	21.5	3.4
Mental health-related	75.3	61.6	54.9	...	34.9	24.6	27.8	38.6	35.9	8.7
Dexterity	86.1	82.1	75.2	39.5	...	28.7	31.3	25.9	29.7	5.5
Hearing	67.3	58.5	56.2	30.1	31.3	...	30.2	21.0	26.8	4.5
Seeing	74.1	66.7	64.0	39.0	39.3	34.9	...	28.0	30.5	6.2
Learning	74.1	65.2	59.4	66.2	39.7	29.5	34.0	...	53.6	16.7
Memory	80.2	71.3	67.5	61.6	44.8	37.2	36.9	52.9	...	9.9
Developmental	49.2	48.3	41.8	57.1	32.2	24.5	28.9	64.2	39.0	...

2.1.1 Government regulations and plans

Through Accessibility for Ontarians with Disabilities Act (Government of Ontario, 2005), the Government of Ontario has aimed to make all private and public spaces in Ontario accessible by 2025. It sets out process for developing and enforcing accessibility standards that every public and private organization should meet. The standards have been categorized into:

a. **Customer Service Standard**: Customer Service Standard consists of a set of regulations that ensures that the goods, service or facilities provided by an organization are served in a manner that respects the dignity and independence of person with disability and ensures that they get the same opportunity to access the service as any other person would get. It also ensures that the service premise allows access to service animals or support person.

b. **Information and Communication Standard:** The information and communication standard ensures that a person with disability has access to information provided by an organization in a format that is accessible at no additional cost. This applies to all the information provided by organization including the training resources, web site and materials published online.

c. **Transportation standard:** The transportation standard ensures that all the transportation service providers make the information about accessibility features of their vehicles available to public. It also ensures that people with a disability should be provided with needed accommodation while they are on the vehicles with no additional cost.

d. **Employment Standard:** The employment standard ensures the rights of people with disabilities during the hiring and selection process. It also ensures that the employees with disabilities are provided with needed accommodation at the work place.

e. **Design of Public space standard:** This standard applies to all the public spaces maintained by government or public organizations such as recreational trails, beach access routes, outdoor picnic area and playground, parking, etc. The standard mandates that the space can be used by people with disabilities such as people using mobility equipment. It includes the policies for minimum width of trail, design of entrance, signage and information, slope of trail and wheelchair ramp, accessible washrooms, etc. This ensures that people with accessibility need would face minimum physical barriers while visiting such place.

2.2 Recommendation Systems

2.2.1 Introduction

Recommendation Systems are software tools and techniques that help people make choices by presenting them with suggestions based on the experience of other users. (Resnick & Hal R., Recommender systems, 1997) (Ricci, Lior, & Bracha, 2011) With the increase of publicly available information and choices, recommendation systems have gained significant popularity on both industry and academia and have been widely used on the internet by e-commerce websites, music and media services, news, social networks etc. to promote the sales as well as help users find interesting items.

Recommendation algorithms are used as a tool for personalization so that the products and services offered to a user during his interaction with the service are filtered according to his interest (Linden, Smith, & York, 2003). This ensures that the user would have a better experience finding interesting goods and services and the service will benefit from the increased sale.

Recommendation systems may use implicit, explicit or both types of feedback as the data required to generate a recommendation. Explicit feedback is collected by asking users to directly rate an item or service they used (Pazzani & Billsus, 2007). An example of explicit feedback is rating the product purchased in an e-commerce system. Implicit feedback is the data collected through the use of the system without asking them to rate them (Oard & Kim, 1998). For example, if the user adds an item to wishlist, it means that the user liked that item. While explicit ratings are more reliable and less noisy, users are less likely to explicitly rate each item they interact with (Pazzani & Billsus, 2007).

2.2.2 Recommendation Approaches

Recommendation Systems can be broadly classified into three types based on the techniques used to recommend items to the user:

2.2.2.1 Collaborative Filtering (CF):

CF is the recommendation technique based on the principle that people like the product/services liked by people similar to them. It uses algorithms to find the unknown preference of a product or service to a user based on the known preference for the same product by similar users (Su & Khoshgoftaar, 2009).

Memory-Based CF algorithms use statistical methods to find a set of neighboring users selected based on the similarity between them (Su & Khoshgoftaar, 2009). The similarity between users is high if their historical ratings for similar product agree and low if the ratings do not agree. Once the neighborhood is found, the rating for an unknown item for a user is calculated by aggregating the ratings for the same items by the users in his neighborhood.

Memory-based CF algorithms use a database of user-item preference matrix that consists of a list of users as rows and a list of items as columns. Each entry represents whether the user likes or dislikes the item in some form of rating scale like 0-5, like-neutral-dislike, etc. Algorithms are used to predict the probability that the user would like any additional item that he has not rated. For example, table 2 represents a user-item matrix with three users and four items:

Table 2: An example of a user-item matrix

	I_1	I_2	I_3	I_4
U_1	4	?	2	4
U_2	?	5	3	?
U_3	?	3	2	3

CF algorithm finds the predicted rating of an unrated item using the following steps (Sarwar, Karypis, Konstan, & Riedl, 2001):

- Calculate similarity weight $sim(u, v)$ between the user u and v : Different similarity measures such as cosine-based similarity, correlation based similarity (Sarwar, Karypis, Konstan, & Riedl, 2001) can be used to calculate the similarity.
    - Pearson correlation is calculated by first identifying the set of items I_{uv} rated by both u and v.

Equation 1: Pearson Correlation

$$sim(u,v) = \frac{\sum_{i \in I_{uv}} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_{uv}} (r_{v,i} - \bar{r}_v)^2}}$$

Where $r_{u,i}$ is the rating for item i by user u and $\bar{r}_u$ is the average ratings by user u for items rated by both the users. For example, the similarity between users 1 and 3 in example table 2 is $sim(U_1, U_3)$ = 0.71.

- The cosine-based similarity between user u and v can be calculated as:

Equation 2: Cosine Similarity

$$sim(u,v) = \cos(\vec{u},\vec{v}) = \frac{\vec{u}\cdot\vec{v}}{\|\vec{u}\| \times \|\vec{v}\|} = \frac{\sum_{i\in I_{uv}} r_{u,i} r_{v,i}}{\sqrt{\sum_{i\in I_u} {r_{u,i}}^2}\sqrt{\sum_{i\in I_v} {r_{v,i}}^2}}$$

Where I_{uv} is the set of items rated by both user u and v and $r_{u,i}$ is the rating for item i by user u.

- Predict unknown rating: Once a set of neighbors $u \in U$ have been decided based on the similarity weight defined above, the predicted rating $P(a,i)$ for user 'u' and item 'i' is calculated as (Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994):

Equation 3: Predicted Rating

$$P(a,i) = \bar{r}_a + \frac{\sum_{u\in U}\left(r_{u,i} - \bar{r}_u\right)\cdot sim(a,u)}{\sum_{u\in U} |sim(a,u)|}$$

Where $\bar{r}_a$ and $\bar{r}_u$ are the average ratings for all the rated items by users a and u respectively.

- Top-N recommendation: a set of N top-ranked items are generated for recommendation to the user.

Model-Based CF approach creates a summarized model of data using machine learning methods such as Bayesian network, clustering and rule-based approaches (Sarwar, Karypis, Konstan, & Riedl, 2001). The trained model is used to predict the unknown rating and then generate a top-N recommendation (Aggarwal, 2016).

2.2.2.2 Content-Based Filtering

Content-based recommendation systems use an algorithm to analyze the match between a user and an item based on the description of the item and the user profile information (Pazzani & Billsus, 2007). Item description could consist of structured data such as a database of books consisting of a title, author, and publisher, as well as unstructured text consisting of book descriptions, cover image and reviews. Similarly, a user profile could consist of preference and historical interactions by the user with different items (Pazzani & Billsus, 2007). If both user and item profile are complete and accurate, an effective and accurate recommendation can be made to the user. Following are general components being used by content-based recommendation systems (Lops, De Gemmis, & Semeraro, 2011).

- **Content Analyzer**: It performs the pre-processing of unstructured information (such as text description) so that the result can be used as input to the next stage. Feature extraction techniques are used to extract actionable information such as keywords and their frequency. For example, Term-frequency Inverse-document-frequency (TF-IDF) is used to identify the importance of keywords to a given document (Salton, 1989).
- **Profile Learner**: This module uses learning techniques such as clustering, neural networks, and classification algorithms to learn the general preference of the user (Ali, El Desouky, & Saleh, 2016). Details of items liked or disliked in the past is used to infer the interest of the user.
- **Filtering Component**: Filtering Component uses the profile learned by the Profile Learner and the item information extracted by Content Analyzer to find the match between the user's profile and the content. A higher match indicates that the item could be more interesting to the user.

2.2.2.3 Hybrid Recommendation

Hybrid Recommendation System is the combination of two or more recommendation approaches such as collaborative filtering, content-based filtering, data mining techniques and a mathematical model to gain better performance (Burke, 2002). Following are some of the approaches used to combine multiple recommendation techniques:

- **Weighted**: Recommendation generated using multiple techniques are combined using some weights for the result from each technique. For example, the recommendation system proposed by (Santos, Almeida, Martins, Gonçalves, & José, 2017) computes recommendation using multiple criteria and techniques and later combines them using weighted sum.
- **Switching**: The system can switch between methods based on some given situations. For example, if the confidence (predicted rating) generated by the

collaborative filtering technique is less than the set threshold, the system can switch and execute the content-based filtering to generate the recommendation for the user (Burke, 2002).

- **Mixed**: This is a popular technique to combine multiple recommendation techniques in which techniques from two or more approaches are mixed during the recommendation process. For example, a collaborative filtering system can use the keywords extracted from user profiles in addition to the ratings to find the similarity between users.

2.2.3 Evaluation of Recommendation Systems

Different methods have been discussed in the literature for evaluation of recommendation systems (Herlocker, Konstan, Terveen, & Riedl, 2004) (Shani & Gunawardana, 2011) (Schein, Popescul, Ungar, & Pennock, 2002). Following are the major aspects considered for evaluation:

- **Accuracy**: It is the most important aspect for the evaluation of recommendation system. Most of the recommendation system depend on the prediction of utility such as predicted rating, the match between user and items, etc. Prediction accuracy of the system is calculated by comparing the predicted rating from the system with the real rating from the user. Prediction accuracy is calculated offline using either natural or synthesized data set (Herlocker, Konstan, Terveen, & Riedl, 2004). Mean Square Error (MSE) and Root Mean Square Error (RMSE) are the most popular metrics used to evaluate the accuracy of the system. Given a test set T of user-item pairs (u, i) for which the rating r_{ui} by user u for item i are known, computed rating $\hat{r}_{ui}$ by user u for item i is calculated using the algorithm. RMSE is computed as:

Equation 4: Root Mean Square Error (RMSE)

$$RMSE = \sqrt{\frac{1}{|\mathrm{T}|}\sum\nolimits_{(u,i)\in T}(\hat{r}_{ui} - r_{ui})^2}$$

And, MSE is computed as:

Equation 5: Mean Square Error (MSE)

$$MSE = \sqrt{\frac{1}{|\mathrm{T}|}\sum\nolimits_{(u,i)\in T}|\hat{r}_{ui} - r_{ui}|}$$

- **Cold Start**: The recommendation system is evaluated by its ability to address to a new user and items whose preference and relation with other items are unknown (Schein, Popescul, Ungar, & Pennock, 2002). Content-Based filtering algorithms perform better than collaborative filtering techniques when a new user or item is introduced to the system because collaborative filtering depends on the historical preference of a user as well as the ratings received by an item.
- **Diversity**: Suggesting similar items to the user might fill the result with items or products from the same category that are similar to each other. The user would have a hard time finding the product if diverse products are not recommended (Shani & Gunawardana, 2011). For example, if the user is trying to find a restaurant, recommending him five restaurants serving similar cuisine in the same area might not be effective. A good recommendation system should have a balance between accuracy and diversity of result.
- **Utility**: Utility is the measure of value for recommending an item for the recommender system owner (Shani & Gunawardana, 2011). For example: for an e-commerce system, the utility is the profit earned by selling an item. Utility has to be considered along with accuracy and diversity to maximize the profit while giving maximum value to the user.

2.3 Point Of Interest (POI) Recommendation System:

Point of Interest (POI) means any places such as a library, restaurant, hospital, park etc. that people could be interested to visit. This includes business, buildings, public places that can be represented on a map. POI recommendation services help users find new places and help them know their city better [7]. These help people decide the places to visit in their own cities as well as in a new city based on criteria such as preference (He, Li, Liao, Song, & Cheung, 2016), geographical and social influence (Ye, Yin, Lee, & Lee, 2011), temporal information (Quan & Cong, 2013), road conditions (Megen, Grummon, Lobben, Omri, & Perdue, 2017) and user profile (Gao, Tang, Xia, & Liu, 2015).

2.3.1 Factors affecting POI recommendation

- **Successive Places of visit**: It is based on the assumption that the next place people would visit is influenced by the current location of the person (Cheng, Yang, Lyu, & King, 2013) (Chen, Li, Cheung, & Li, 2016). For instance, if a swimmer at an airport has options to go to a water park or a hotel, he will prefer to visit the hotel. Figure 2 below shows examples of different check in sequence for users. Matrix factorization method FPMC-LR (Cheng, Yang, Lyu, & King, 2013) has been used to find next best place for a given user by analyzing the movement pattern of other users. While most of the recommendation methods only consider the transition between POI categories, (Zhao, et al., 2018) proposed Long Short-Term Memory (LSTM) based method to model the Spatio-temporal relationship between check-ins of users and hence make recommendation based on short-term and long-term interest of users.

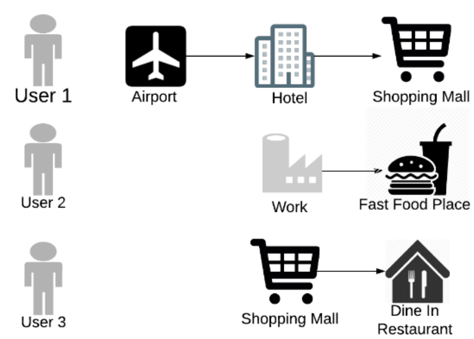

Figure 2: Example of user's check-in sequence

- **Social Influence**: The places a person visits are influenced by his friends and various social groups he belongs to. (Chen, Li, Cheung, & Li, 2016) Having used distance weighted Hyperlink-Induced Topic Search (HITS) algorithm to find the visiting frequency of people in the social group. Also (Song, et al., 2015) analyzed Location Based Social Network (LBSN) data and proposed the probabilistic model to predict next location considering temporal, spatial and social influence.
- **Geographical Influence**: People tend to visit the places near their home or the places near the locations they are considering to visit (Ye, Yin, Lee, & Lee, 2011).

Distance has been used by (Chen, Li, Cheung, & Li, 2016) as a weighting factor to calculate the relation between place and users.

- **User Reviews**: Extracting information from reviews and using it to model users and POIs on various aspects can help generate helpful and explainable recommendation for users. (Baral, Zhu, Iyengar, & Li, 2018) proposed the use of deep neural network to formulate the correlation between reviews and various aspects discussed in it. For example, the review sentence "*though the staffs were not very friendly, the coffee there was really good*" indicates positive sentiment for the food but negative sentiment for customer service; where 'food' and 'customer service' can be two different aspects used for recommendation. With this information, user can know the reason why an item is recommended to him.

2.4 Accessible POI Recommendation

People with disabilities have one or more accessibility needs that has to be fulfilled to enable them to visit the place independently. POI Recommendation system for people with disabilities should be able to recommend places to the people based on the accessibility needs of the users and the accommodations provided by the place (Santos, Almeida, Martins, Gonçalves, & José, 2017). Systems like this help people with disabilities find accessible places to visit independently.

The experiment conducted by (Lyu, 2017) (Lyu, 2017), studied the travel choice of people with disabilities. Based on the responses collected, it was found that people with disabilities care most about the accessibility accommodation facilities while deciding on the place to visit (Lyu, 2017). While extensive research has been conducted on POI recommendation, only a few of them have considered disability of a user and accessibility of places into account (Santos, Almeida, Martins, Gonçalves, & José, 2017).

2.4.1 Crowdsourced information on Accessibility

Crowdsourcing to collect information from a large group of people have been successfully used in a number of ways to help people with disabilities. (Bigham & Ladner, 2011) In addition to POI recommendation, crowdsourcing has been successfully used in the past to collect information about road hazards (Santani, et al., 2015), public health issues (Brabham, Ribisl, Kirchne, & Bernhardt, 2014) and various another application area.

Most of the crowdsourced applications for people with disabilities on the web use maps and provide information layers that show the places being searched with additional information represented by colors or text on whether or not the places are accessible (AXS Map, 2015) (Access Now, 2017). But, none of these applications seem to have utilized the profile information of the user to determine whether the place meets user's accessibility needs. In addition, they do not fulfill all the needs of people with multiple disabilities which is common among people (Statistics Canada, 2012).

2.4.2 Existing POI search and recommendation services on the web for people with disabilities

A number of applications have been developed to collect accessibility information of places from the pool of volunteer users and using those data to help people with disabilities find accessible places (Mobasheri, Deister, & Dieterich, 2017) (AXS Map, 2015) (Megen, Grummon, Lobben, Omri, & Perdue, 2017) (Access Now, 2017) (Access Locator, 2017). While most of these services focus on a single category of disability; for example: finding wheelchair accessible places (Mobasheri, Deister, & Dieterich, 2017) (AXS Map, 2015) (Megen, Grummon, Lobben, Omri, & Perdue, 2017), some of these systems collect and utilize crowdsourced data to help people with multiple disabilities (Access Now, 2017) (Access Locator, 2017). Table 3 shows the comparison of five accessible POI recommendation systems in use based on our experience of using these systems:

Table 3: Existing crowdsourced accessible POI web applications

	Wheel Map (Mobasheri, Deister, & Dieterich, 2017)	**Access Now** (Access Now, 2017)	**Axs Map** (AXS Map, 2015)	**EUG Access** (Megen, Grummon, Lobben, Omri, & Perdue, 2017)	**Access Locator** (Access Locator, 2017)
Purpose	Accessible POI	Accessible POI	Accessible POI	Accessible Routing	Accessible POI
Data Source	Crowdsourced Data	Crowdsourced Data	Crowdsourced Data	Maps, and GIS Data	Crowd Sourced Data
Rating Factors	Wheelchair accessibility, accessible bathroom	Parking, washroom, braille, elevator, quiet, spacious	Wheelchair, Bathroom, Steps (Boolean: sound, parking, light, guide dog)	Crosswalks, curb cuts, pedestrian crossing, elevation	Social/cognitive, vision, hearing, communication, physical disability.
Rating Scale	Accessible, partially accessible, not accessible	Accessible, partially accessible, patio accessible, not accessible	1-5	Yes/No	Yes/No → converted to % based on number of yes.
Quality rating of POI	No	No	No	No	Yes
Reviews	No	No	Yes	No	Yes
Pictures of places	Yes	No	Yes	No	Yes
Multiple disabilities	No	Yes	No	No	Yes
User Profiles	No	No	No	No	Yes
Personalized Results	No	No (Filter by accessibility)	No	No	No
Accessible Routing	No	No	No	Yes	No

2.4.3 Related Work: Using POI functionality and accessibility levels for delivering personalized tourism recommendations

(Santos, Filipe, et al., Using POI functionality and accessibility levels for delivering personalized tourism recommendations, 2017) proposed a recommendation system that uses the physical and psychological limitations of users and POI profiles to recommend places for people with disabilities to visit. The research focuses on modeling of the user and POI profile including the level of functionality (for users) and the measure of accessibility facilities available (for POI). The proposed application has two layers:

Knowledge Layer: The knowledge layer is the representation of models used to represent users and POIs. Users are represented as:

Functionality model: This model represents the intellectual, hearing, vision and locomotion level of the user. This is used to identify if and to what extent the user needs accessibility accommodation based on these needs.

- **Society model**: This is the combination of the Social and Community model to which the user belongs to.
- **Tags model**: Tags model consists of the set of tags used by the user during the interaction with the system and their weight based on the frequency used by the user. This represents the interest of the user.
- **Stereotype model**: This model represents the general interest of the user. It stores the level to which user belongs to the stereotype: Gastronomy, Nature, Business or City breaks.
- **Emotions**: This model represents the user's emotion; whether the user is surprised, happy, angry or sad, while he is at different classes of POIs. POI classes could be monuments, parks, etc.

Reasoning Layer: The reasoning layer consists of a hybrid recommendation system that considers accessibility, tags, and the stereotype of users and POI to generate a list of recommended POIs for the user.

- **Accessibility recommendation model**: This is based on the relation between user's need for accessibility and the POI's accessibility profile. For example: if the user's vision need is 0.6, the hearing need is 0.7 and the building's accessibility level for vision is 0.8 and hearing is 0.7, the accessibility level of the building for the user would be 0.6 X 0.8 + 0.7 X 0.7 = 0.97. Available POIs are sorted based on the accessibility level to generate a recommendation.
- **Emotion-based recommendation model**: This is based on the emotion reaction detected by the system when the user was shown different pictures of places representing different POI classes.
- **Tags based recommendation**: This is based on the weight of tags in user's profile and the same tags in the POI profile. Higher match results in the place being at top of recommendation list.

- **Social Network based recommendation model**: This integrates friend's preferences on the recommendation of each user. The paper assumes that if the social circle of a user is interested in a place, it is most likely that this user would also like it.

The weighted sum was used to compute the final recommendation from the above techniques. This makes sure that the recommended places interest the user as well as fulfills the accessibility requirements.

2.5 Rating Aggregation

The internet has a huge amount of goods and services to offer to its users. But, unlike making choices by observing, feeling or using the goods or services, users have to rely on the information available online to make choices. Therefore, reputation and trust-based feedback mechanisms have been used widely in online communities in the form of ratings and reviews (Josang, Ismail, & Boyd, 2007). While reviews contain qualitative feedback from the user in terms of textual description, images, videos, or their combination, ratings are quantitative feedback in which the user rates the item offered within a given scale. The rating could represent the overall satisfaction of the user or could represent their opinion on a specific aspect (Josang, Ismail, & Boyd, 2007).

Webster dictionary defines aggregate as "*a whole formed by combining several (typically disparate) elements*". The aggregate of series of ratings given by multiple users at different point of time is a value typically represented out of 5 stars or as a percentage value, which represents overall opinion of people for the item across that timeline. Figure 3 demonstrates five star ratings used in Google Maps and Figure 4 demonstrates the binary ratings on different criteria of a place. When people are presented with an item on the internet, the aggregate of ratings presented helps influence their decision (Chintagunta, Gopinath, & Venkataraman, 2010).

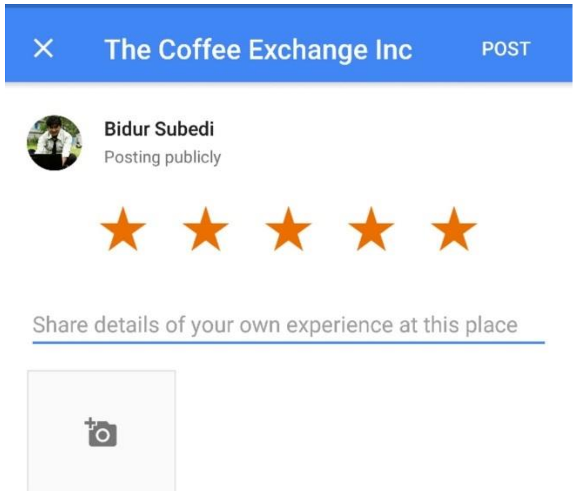

Figure 3: Overall satisfaction rating on Google Map

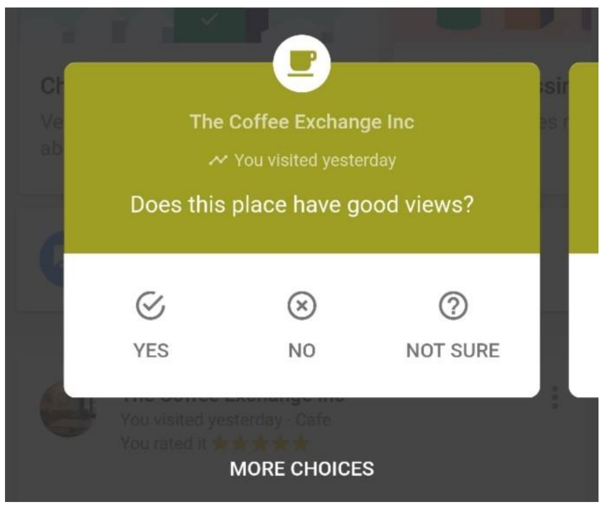

Figure 4: Specific Issue rating on Google Map

Different aggregate techniques such as mean, median, mode, etc. could be used depending on the nature of data and the purpose of aggregate. We are only considering binary ratings in which the user chooses between two options (true or false) while discussing the aggregation techniques. The first sub-section discusses popular statistical techniques used for rating aggregation while the second sub-section discusses the techniques for calculating temporal aggregate that represents the state of goods or services at given point of time using a stream of ratings.

2.5.1 Rating Aggregation Techniques:

2.5.1.1 Mean as the aggregate value

Mean rating has been used in most of the e-commerce and review collection websites to represent the aggregate rating. Given a series of positive (true) and negative (false) rating, if each positive rating is represented as 1 and negative rating is represented by 0, the mean of binary ratings is calculated as:

Mean Rating=$\frac{Sum\ of\ positive\ reviews}{number\ of\ reviews}$

If we consider all the rating to be equally important, and we have the following rating stream for the place 'p_1' and criteria 'c_1':

False, False, False, False, True, True, True

The mean aggregate of these ratings where True is considered 1 and False is considered as 0.

$S(p_1,c_1)= \frac{0+0+0+0+1+1+1}{7}$=0.43

This represents a 43% confidence that the place actually fulfills the accessibility criteria 'c_1'. While this approach is simple and would be suitable for static items such as 'movie', 'gadget', etc. it is not very efficient for this application as the places are constantly changing. The place might have fixed the lighting since the last user rated.

2.5.1.2 Voting as the measure of aggregate

Voting as the measure of aggregate value considers the most repeated rating to be the representative (aggregate) rating of the criteria for a place. If we have the following rating stream for the place 'p_1' and criteria 'c_1':

False, False, False, False, True, True, True

Here, we have the following frequency counts:

Table 4: Votes for ratings for example data

Rating	Frequency
True	3
False	4

Since False is repeated a maximum number of times, voting would return 0% confidence that the place fulfills the criteria. But, the place could have bad lighting for a long period of time followed by improved lighting for the last few months. Voting in such situation does not properly represent the state of the criteria at a given time.

2.5.2 Temporal Aggregation Techniques

Given a series of ratings received over a period of time, the overall aggregate represents the status of the item over that period of time. Though it is a useful indicator of the overall opinion of users, the state of the item could change over the period of time, and the aggregate may or may not accurately represent the quality of item or opinion of the user at current situation (Ding & Li, 2005). So, the temporal aggregate of the rating stream is the value that most likely represents the quality or opinion of users towards the item at a given point in time.

A good temporal aggregate for the rating stream should have a minimum error (fluctuation from actual state or opinion it represents) as well as it should adapt to changes in actual state or opinion. The aggregated rating should:

- Represent all the ratings given by the user.
- Be sensitive to change in the condition or quality of accessibility accommodation at the place.

2.5.2.1 Sliding Window Aggregation

Sliding Window is a window of last n ratings we've received, where n is a parameter that represents the number of latest ratings we need to observe in order to determine the aggregate. The value of n might be different for different criteria.

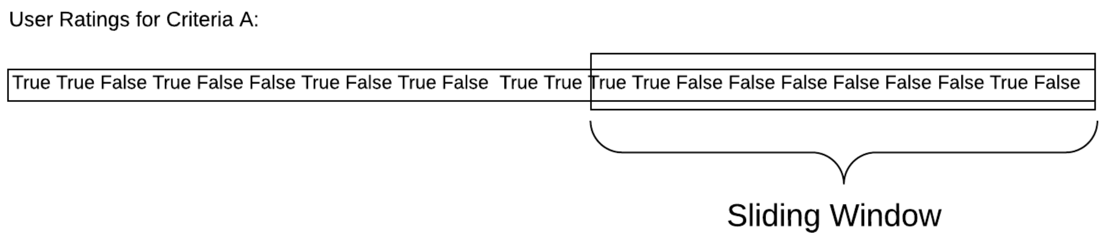

Figure 5: Confidence Window of last 10 ratings

For example, if the figure above represents a stream of positive or negative ratings for an item, and the window size is 10, we would only consider the latest 10 ratings while applying the aggregate function. If we consider mean to be the aggregate function, the aggregate for this window would be:

mean(True, True, False, False, False, False, False, False, True, False)

Similarly, other aggregate function such as voting could also be used in a similar way using a sliding window. Since this technique only considers a limited number of ratings at a time, it easily adapts to the changes in the quality of place or change of opinion of users. For example, figure 6 shows constant deviation in aggregate using sliding window mean.

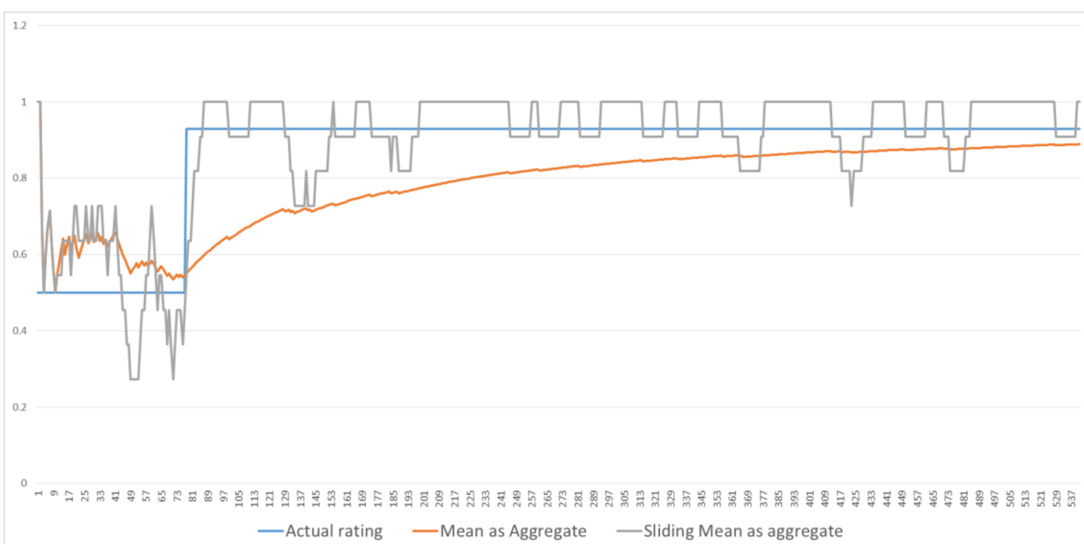

Figure 6: Constant error on Sliding aggregate (window size=10)

Deciding the window size is a challenging problem while using sliding window aggregate. Smaller window size help adapt to changes quickly and a larger window size have higher accuracy as more number of samples are used. So, window size should be chosen to get acceptable accuracy and adaptiveness. Since the number of ratings (n) considered on each aggregation is constant, even if we have high number of overall ratings, the aggregation error remains constant.

2.5.2.2 Time Weighted Aggregate

Damped/time fading window model (Ding & Li, 2005) is used to calculate the average confidence score of the ratings. In this model, the weight of old data fades while the latest data has the highest weight. This is to ensure that the confidence score represents the current status of the place. The damping function/time weight function is defined by:

Equation 6: Time Weight Function

$$f(\Delta t) = \mathrm{e}^{-\lambda\,\Delta \mathrm{t}}$$

Where $\Delta \mathrm{t} = t_{latest} - t_{this}$ for the rating represents the duration between the times this rating was received and the time latest rating was received. λ is the rate of decay of time weight of rating. Higher value of λ assigns low weight to historical data while lower value of λ assigns higher value to historical data. If T_0 is the half-life; that is the weight reduces by half in T_0 days, the rate of decay λ is defined by:

Equation 7: Half-life Function

$$\lambda = \frac{1}{T_0}$$

So, the time weight depends on the value of the half-life parameter T_0. Graph on Figure 7 represents the curve of time functions for different values of T_0.

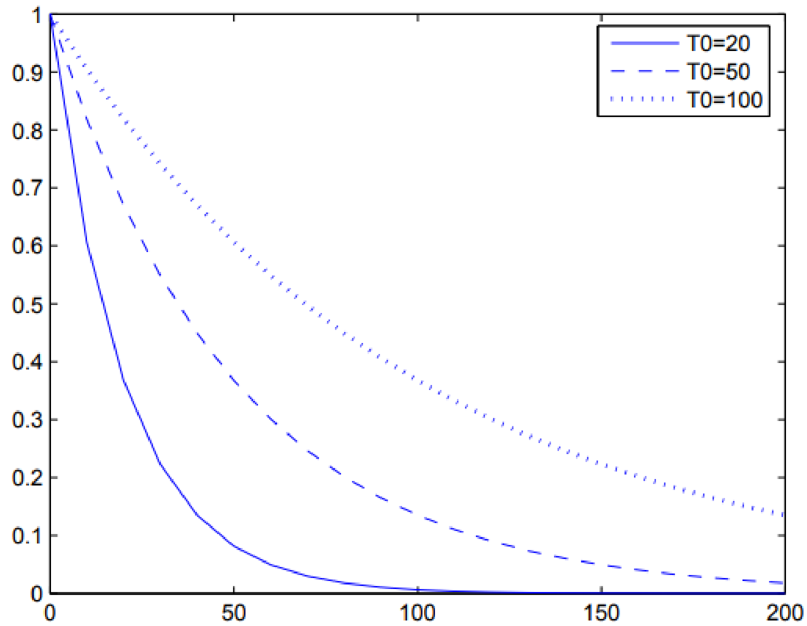

Figure 7: Graph representing the curve of functions for different T_0 values (Ding & Li, 2005)

This means a low value of T_0 reduces the impact of historical data, while a higher value increases the impact of historical data. In order to calculate the confidence score that represents the current state of the place, we need to determine the appropriate value for the parameter T_0.

If the values of latest ratings are consistent, we should have a shorter half-life so that the current consistent ratings have the higher impact of the calculated confidence score. But, on the other hand, if the latest ratings are inconsistent, this means, we cannot rely on just the latest data to calculate the score. In this case, we should have a longer half-life to account for the historical data to calculate the aggregate value.

Ratings for Criteria A:

True True False True False False True False True False True True True True False False False False False False True False

Ratings for Criteria B:

True True True True False True True True True True False False True False False False False False False False False False

Figure 8: Example of rating streams for different criteria.

For example, in the data represented in Figure 3, ratings in criteria B are more consistent than those in criteria A. This means, users are more confident while rating criteria B, than rating criteria A. For B, we would be more confident to use the latest data to determine how accessible the place is for that criterion, whereas we should consider more historical data for A, as the user seems to be confused about their rating.

CHAPTER 3:
PROPOSED SYSTEM

3.1 Overview

This section is divided into three sections. The first section describes the knowledge layer of the system. The knowledge layer is the data layer that stores the data needed for the recommendation process. The user profile contains information about the location and preference of the user. The POI Database contains a list of Point of Interests (POIs), with their geo-coordinates. User ratings consist of a stream of ratings given by users for different POIs at different points of time.

The second section describes the rating aggregation system and the intermediate database created by this system. Rating aggregation layer aggregates the user ratings about different accessibility criteria to compute an aggregate that best represents the current situation. The aggregated rating along with user's information is used for recommending the places. So, having an accurate aggregate that represents the current state of accessibility criteria of the place is crucial to generating a useful recommendation.

The third section describes the recommendation layer of the system. This layer computes the utility of each POI for the user based on three criteria; accessibility, interest, and distance. When sorted using the utility computed, the system can create an ordered list of POIs that are most accessible to the current user.

Figure 9 below illustrates the relationship between these layers and shows the flow of information between its components:

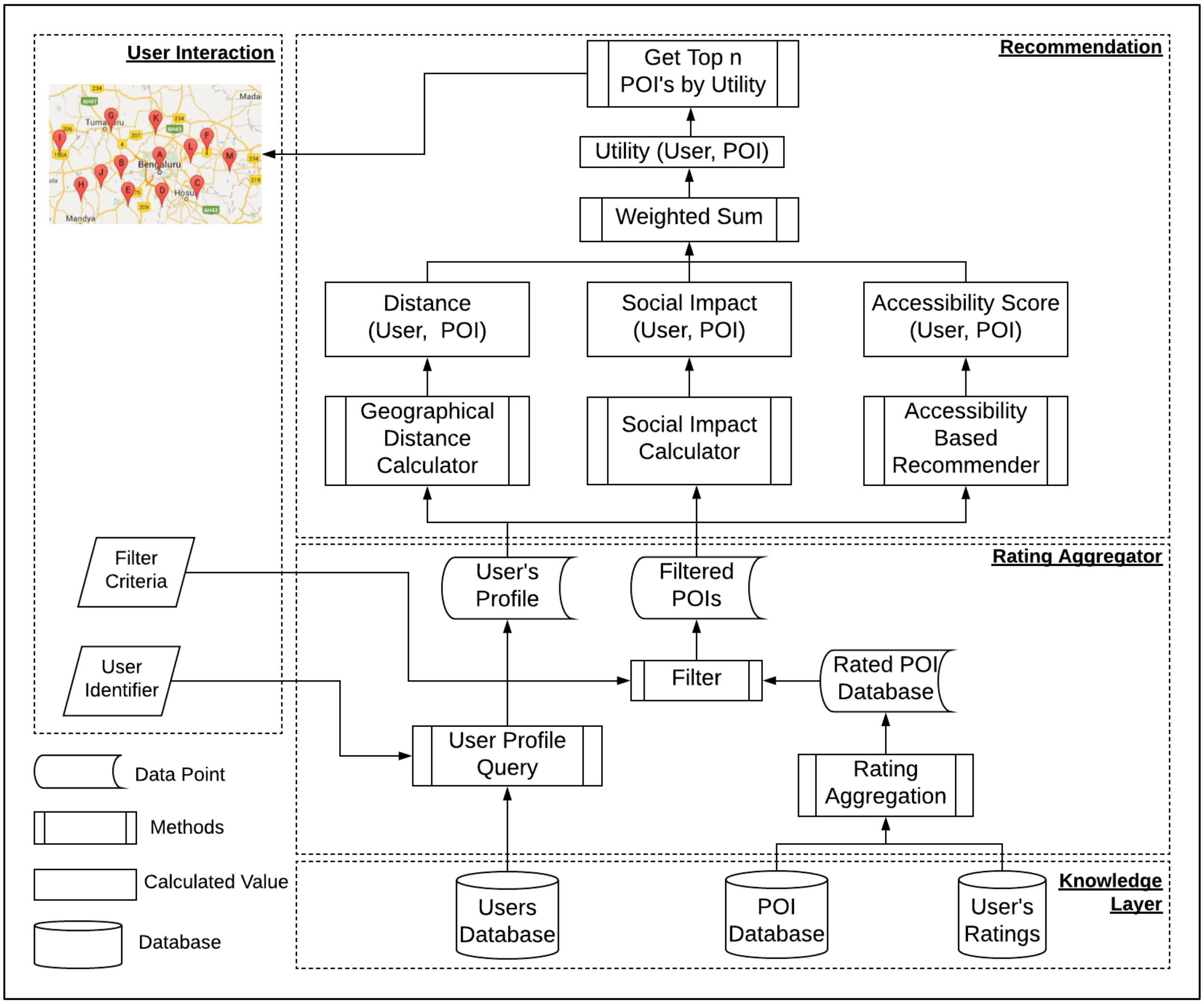

Figure 9: Proposed Architecture of recommendation system

3.2 Knowledge Layer

Knowledge layer stores the data needed for the recommendation. This is composed of the user profile, POI database, and User ratings. The user profile is created when any user signs up to the system. It consists of basic information and accessibility requirements of the user. POI Database is the database of places in the system. It consists of basic information like name and category of the place as well as its geo-location represented by latitude, longitude pair. User ratings consist of the ratings collected from the user for POIs based on accessibility criteria and service provided. These components are shown on Figure 10.

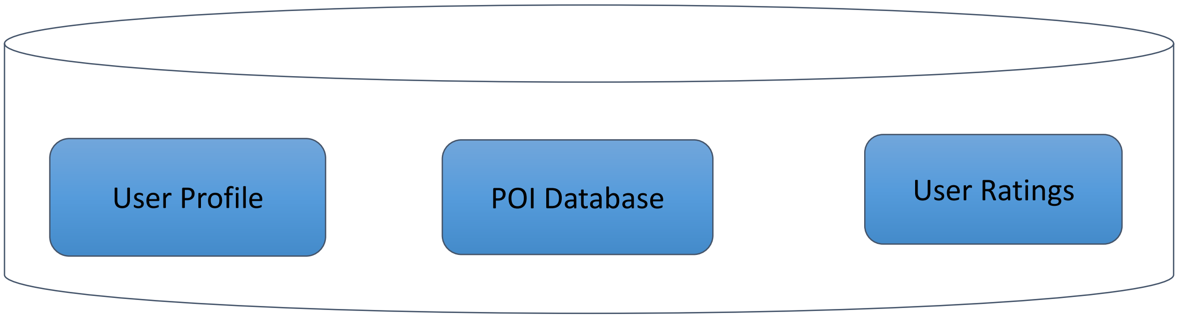

Figure 10 Components of Knowledge Layer

3.2.1 User Profile

We would have a set of users, $\boldsymbol{U_1}, \boldsymbol{U_2} \ldots \boldsymbol{U_m}$. Each user's profile would be represented by a set of accessibility criteria s/he needs fulfilled in order to go to any point of interest (POI). Eg. 'needs parking within 50m of entrance', 'needs ramps leading to entrance', 'needs information/signs in braille', etc. So, $\boldsymbol{C} = \{ \boldsymbol{C_1}, \boldsymbol{C_2}, \boldsymbol{C_3}, \ldots \boldsymbol{C_n} \}$ represents a list of accessibility criteria. Table 5 shows examples of different user profiles with their preference on accessibility criteria.

Table 5: Example of user profile

	$\boldsymbol{C_1}$	$\boldsymbol{C_2}$	$\boldsymbol{C_3}$	$\boldsymbol{C_4}$	$\boldsymbol{C_5}$	$\boldsymbol{C_6}$	..	$\boldsymbol{C_n}$
$\boldsymbol{U_1}$	T	T	F	F	F	T		T
$\boldsymbol{U_2}$	F	F	T	T	F	F		F
$\boldsymbol{U_3}$	F	T	T	T	T	T		F
..								
$\boldsymbol{U_m}$	F	F	T	F	F	F		T

For a user i, and accessibility criteria j, C_{ij} =T means, that user i cares about the accessibility criteria i, and C_{ij}=F means that the user does not care about that criteria. A person can have one or more disabilities and can care about one or more different accessibility criteria.

3.2.2 POI Database

Point of interest is any place eg. Restaurant, park, library, etc. that a person might want to visit. A point of interest has a name, category, and geo coordinates:

- **Place ID**: This is the identifier for the POI. It is automatically generated (incremented) when new places are added to the system.
- **Name:** The name of the POI being displayed to the user.
- **Category**: Category is the numeric value that represents the class such as educational institution, restaurants, and coffee shop that the POI is categorized into.
- **Latitude**: Latitude represents the angle, measured in degrees above or below the equator (Stern, 2004). It along with the longitude represents a position on the earth. Latitude is in the range of -90° and $+90^{\circ}$.
- **Longitude**: Longitude represents the angle, measured in degrees to the east or west of the prime meridian passing through the Royal Astronomical Observatory, England (Stern, 2004). Longitude is in the range of -180° and $+180^{\circ}$. Longitude paired with Latitude is used to represent a position on earth.

Table 6: Example of POI Database

Place ID	Name	Category	Latitude	Longitude
1	POI1	C1	31.2215	-52.5661
2	POI2	C2	22.6665	28.6665
3	POI3	C3	88.5255	-22.6652
…				
n	POINT	CN		

3.2.3 User Ratings

Once a user visits any of the POIs, s/he will give explicit feedback on whether or not the POI is accessible based on the criteria, $\boldsymbol{C}= \{\boldsymbol{C_1},\boldsymbol{C_2},\boldsymbol{C_3},\dots \boldsymbol{C_n}\}$. These are the same criteria used for user profile creation. For the POI, for each criteria, the user would respond on whether the place fulfills that accessibility need. The user may not respond for all criteria or respond to them as ‘not applicable for the place’ or ‘did not look’. In addition, we would also have overall satisfaction on ratings from the data for that place. That would be based on the service/product and overall satisfaction the user had on visiting that place.

Table 7: Example User Rating Data

			Accessibility Criteria				
user_id	POI_id	Date	C_1	C_2	C_3	..	C_n
001	005	<date>	T	T	N/A		F
003	012	<date>	F	T	T		N/A
252	258	<date>	T		T		F
…							
225	288	<date>	T	T	F		T

3.3 Rating Aggregator Layer: Adaptive Time Fading Aggregate

When a user visits a place and rates it for a number of accessibility factors, they will mark each of them as True or False; i.e. whether the place fulfills that accessibility need. For instance, user A might rate the criteria ‘Adequate Lighting at Parking Lot” as True but user B might think that the light is not adequate and rate it as False. We consider these ratings as a stream of Boolean data.

Using this rating data, we calculate the confidence score ‘S(p,c)’ for the place ‘p’ and criteria ‘c’ to represent the confidence that the place fulfills that accessibility criterion. This is the score in the range of 0 to 1 where 0 represents the lowest confidence, meaning the place does not fulfill the accessibility need and 1 represents the highest confidence, which means the place fulfills the accessibility need. Here, the confidence score S should represent the current state of accessibility criteria. So, it should be an adaptive aggregate based on the rating stream and should change when the opinion of people change. For example, if a place recently built a ramp, its rating series would be:

False, False, False, False, False, False, True, True, True

The system should have high confidence regarding a ramp for above rating series because it recently built the ramp and recent opinion of users is positive.

The proposed adaptive time fading aggregate computes the aggregate on two phases as illustrated on Figure 11. On the first phase, it uses time weighted mean using the half-life value (T_0) computed earlier. And, on the second phase, it adjusts the value of T_0 using a tunnel algorithm (Gorawski, Gorawska, & Pasterak, 2017) so that the next aggregate adapts to the changes, if any.

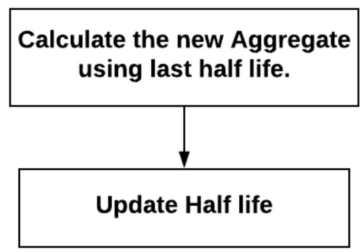

Figure 11: Two phases of Adaptive Aggregation

3.3.1 Damped Aggregate Computation

It is the weighted average which considers all the ratings where the weight of a past rating fades exponentially. The damping function/time weight function is defined by:

Equation 8: Time Weight Function

$$f(\Delta t) = e^{-\lambda \Delta t}$$

Where, $\Delta t = t_{latest} - t_{this}$ for the rating represents the duration or the number of ratings between the times this rating was received and the time latest rating was received. The λ is the rate of decay of time weight of rating.

Equation 9: Decay rate as a function of Half Life

$$\lambda = \frac{1}{T_0}$$

Where, T_0 (half-life) represents the time taken for the weight of the ratings to fall to half of its original value. The initial value of T_0 is set to a high number. When a rating is received, last T_0 is used to calculate the aggregate, and then the T_0 is updated.

So, for n ratings represented by r(1), r(2)…r(n), damped aggregate is calculated as:

Equation 10: Damped aggregate function

$$C = \frac{\sum_{t=0}^{n} r(t) f(n-t)}{\sum_{t=0}^{n} f(n-t)}$$

Where, f represents the time weight function as defined in Equation 8. Here, the numerator represents the weighted sum of all ratings and the denominator represents the sum of weight.

3.3.2 Updating the half-life (T_0)

As the nature of binary ratings allows each rating to be either true or false, for each rating, there are two mutually exclusive outcomes possible. So, the ratings can be represented using Binomial distribution.

Given p, the probability of "true(1)" and (1-p), the probability of having a "false(0)", if no weight is applied, the mean of the rating stream should ideally be p. For such a binomial distribution, the standard deviation of mean is given by:

Equation 11: Standard deviation of mean of Binomial Distribution

$$\sigma_{\bar{x}}(p,n) = \sqrt{\frac{p(1-p)}{n}}$$

Since this is inversely proportional to 'n', the number of ratings considered, with a higher value of n, the variance of mean would be low, and hence the mean would be closer to the actual probability. Maximum error of the mean 'E' is the maximum absolute difference between the mean computed 'μ' and the actual probability 'p' such that $|p - \mu| < \mathrm{E}$ is

Equation 12: Maximum error in

$$E(z,p,n) = z\,\sigma_{\bar{x}}(p,n)$$

given by:

- z: z-score (standard score) for a given confidence level,
- μ: mean of ratings
- p: actual probability/confidence for the given criteria

The objective of adjusting the Half-life (T_0) value is to have a model that adapts well to changes as well as minimizes the aggregation error. Lower T_0 helps adapt to change but increases the deviation. Higher T_0 decreases the deviation (error) of mean but is not able to adapt to change of probability quickly. So, the algorithm helps us detect if there has been a change of opinion (the base probability 'p') from the rating stream and increase or decrease T_0 to obtain maximum possible accuracy and adaptability.

- If the probability is uniform, increase the T_0 (decrease the rate of decay).
- If the probability changes, decrease the T_0 (increase the rate of decay).

We want the past ratings to decay faster if they are irrelevant/inaccurate, and we want the past ratings to decay slower, if they are relevant and could contribute to accuracy. So, detecting the change in the probability (p) helps update the value T0. Following figure represents the expected change in the value of T_0 with the change of base probability:

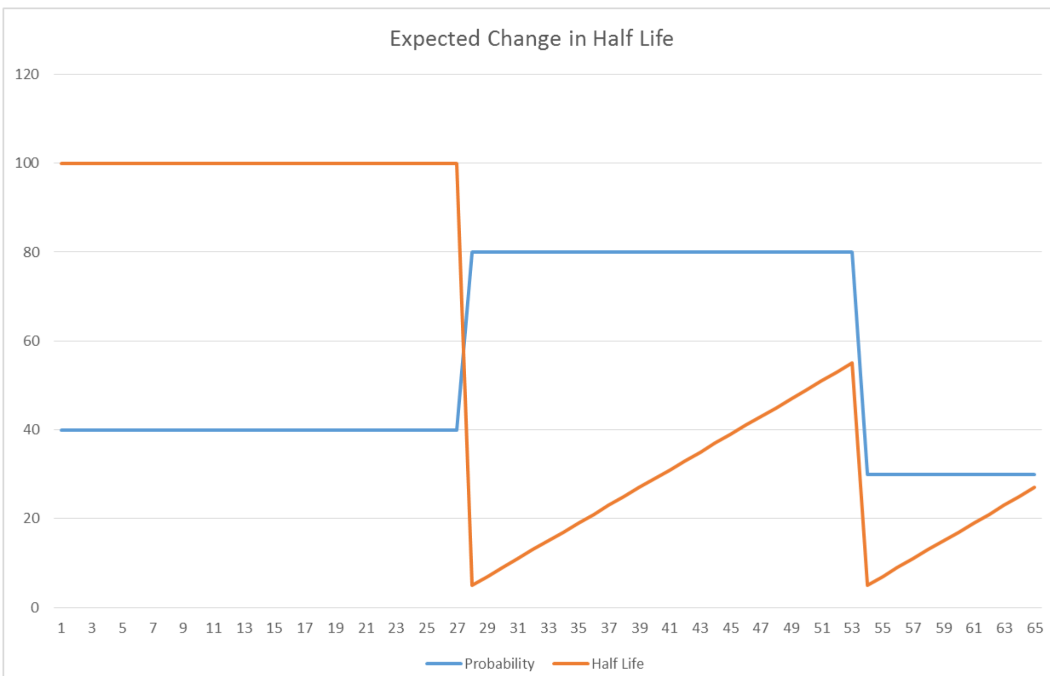

Figure 12: Expected change in half life

The following algorithm is based on (Gorawski, Gorawska, & Pasterak, 2017) for binary data to update the value of T_0:

UpdateHalfLife(n, z, C):

1. $\overline{C_d}$ is the average of last n confidence calculated using (3).
2. Maximum error $E(C_d, n, z)$ is calculated using (5)
3. Acceptable aggregate bound is $(\overline{C_d} - E, \overline{C_d} + E)$
4. If C% of last n sliding window ratings lie within the acceptable aggregate bound:
    - Increase the half-life by 0.15
5. If C% of last n sliding window ratings lie outside the acceptable aggregate bound:
    - Decrease the half-life to 3.
6. If 4 and 5 are not true, it is inconclusive. So, no change of half-life.

We use a window of last 'n' aggregate calculated using the adaptive algorithm and calculate the average of those as the base probability at that point. For instance, in figure 13, a window size of 15 is used to calculate the base probability:

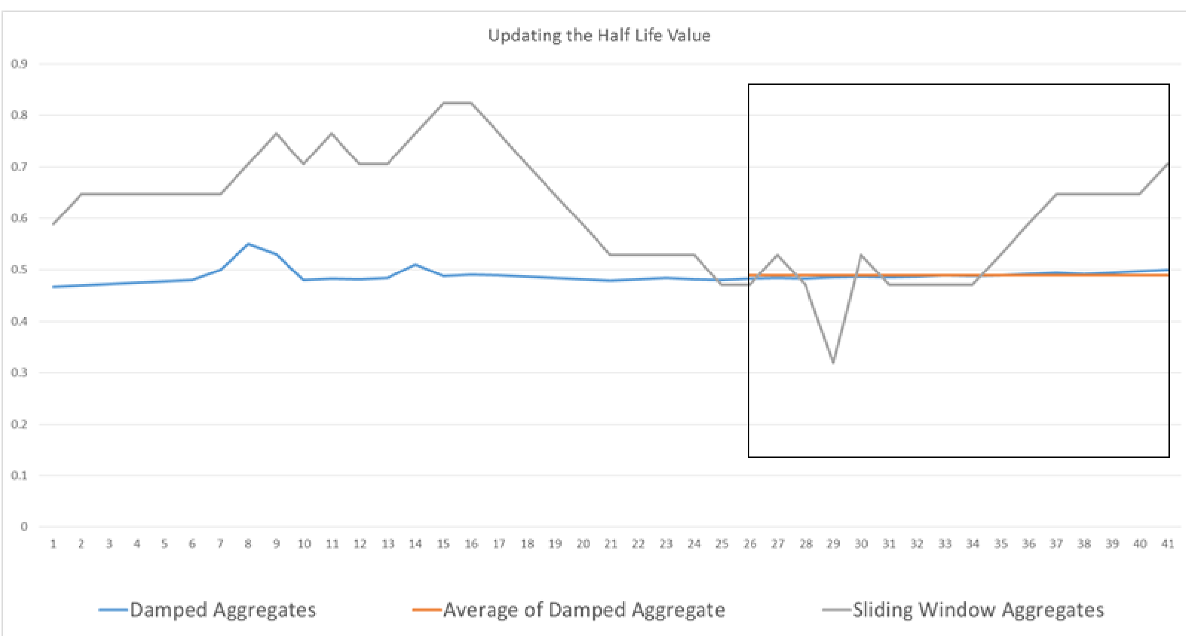

Figure 13: Calculating base probability as window average

The next step is to develop a window of confidence that determines whether the base probability ‘p’ has changed. This window forms a tunnel of maximum and minimum deviation allowed in probability without triggering change of T_0. Figure 14 shows the error bars:

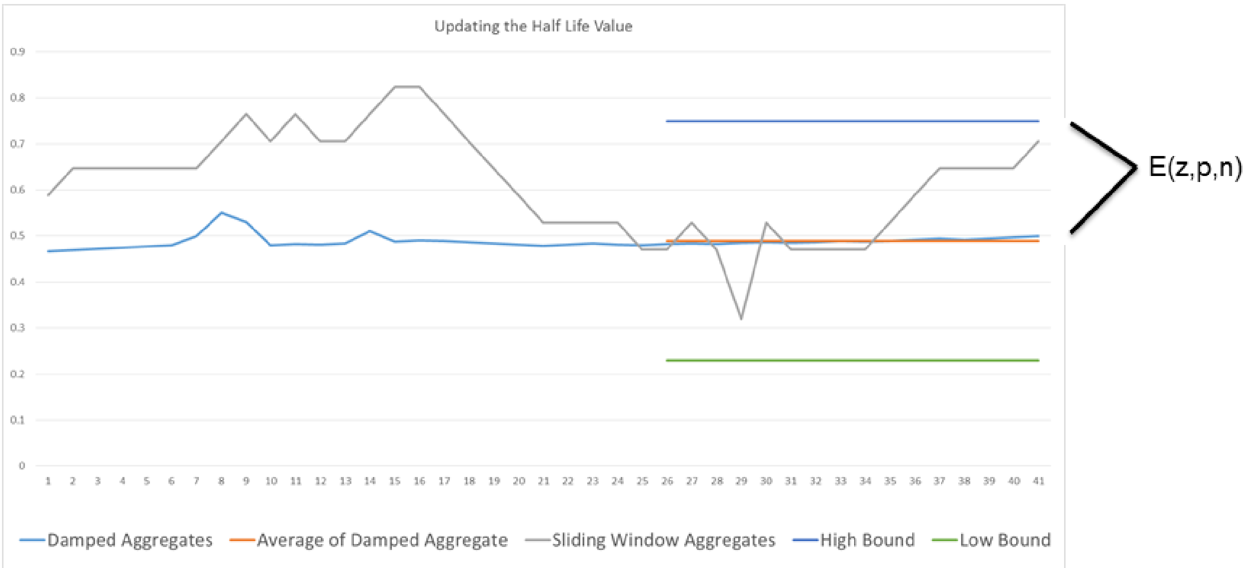

Figure 14: Error bars as tunnel

If the computed aggregate is within the acceptable error bound computed, we would know that the base probability have not changed and hence we increase the T_0 to increase the accuracy of aggregation. But, on the other hand, if the computed aggregate is outside of the error bound, we know that the value of ‘p’ has changed. And, hence, we reset the value of T_0 to minimum to adapt to the change of probability.

3.3.3 Algorithmic Complexity

Given n previous ratings, if a new rating is added, we calculate the new aggregate and update the decay rate.

Calculating new aggregate is a linear time operation as we would compute the aggregate of all available ratings using a constant weight. So, the complexity of calculating the aggregate using equation 3 is O(n).

Once the aggregate is calculated, we update the T_0 to be used for the next rating. As mentioned above, the updated algorithm uses a fixed size confidence window each time, and computes the error bound. So, irrespective of the number of ratings available, the update process is a constant time operation with the complexity of O(1).

So, the overall complexity of computing the aggregate and updating the T_0 is O(n) for each additional rating added to the stream.

3.4 Accessibility based Recommendation System

An accessibility based recommendation system should be able to recommend top-n POIs to each user based on their accessibility criteria and the accessibility level of various factors at that place. **Accessibility Rating** R(p,c) is a function that represents the accessibility of a place (p) on the accessibility criteria (c). This value is in the range of 0 and 1 such that a rating close to 0 represents that the place is not accessible in that criteria and a value close to 1 represents that the place is highly accessible in that criteria.

For each place, and criteria, the rating R(p,c) is the current aggregated value given by the rating aggregation algorithm described in section 3.3. This ensures that the current rating used reflects the current situation of the place for given accessibility criteria using feedback given by other users.

User's Preference: P(u,c) Given a user u and a criteria c, the user's accessibility preference represents whether the given user cares about the criteria. This comes from the profile of the user and entered by user when they create their profile for the first time. The preference value could be one of the following:

- 1 if the user cares about the criteria 'c'
- 0 if the user does not care about the criteria 'c'

Absolute Accessibility Score: S_a(p) of a POI (P_n) is the average of accessibility rating on all accessibility criteria that applies to the place. Though this value represents the overall accessibility of the place, it is not useful for every user as they have different needs. Following is the pseudocode to calculate the absolute accessibility score of a place. If c_1, c_2, c_3 … c_n represents N accessibility criteria,

$$s_a(p) = \frac{1}{N}\sum_{n=1}^{N} R(p, c_n)$$

```
Absolute Accessibility: S_a(p)
total=0
count=0
for each criterias as c_n:
        if R<p, c_n> != N/A:
                count=count+1
                total=total+ R<p, c_n>
return (total/count)
```

Relative Accessibility Score: S_r(p,u) of a POI P_n for a user u is the average of accessibility ratings of the accessibility criteria that the user u_m is concerned about. This represents the personalized accessibility score of the place for the given user. If c_1, c_2, c_3 … c_n represents N accessibility criteria,

$$s_r(p, u) = \frac{1}{N}\sum_{n=1}^{N} R(p, c_n) * P(u, c_n)$$

The following is the pseudocode for calculation of the relative accessibility score:

```
Relative Accessibility: S_r(p, u)
total=0
count=0
for each criterias as c_n:
        if R(p, c_n) != N/A:
                count=count + P(u, c_n)
                total=total+ R(p, c_n) * P(u, c_n)
return (total/count)
```

The relative accessibility score for a user for each place is the profit function for top-n recommendation of the place. So, all the POIs are sorted in descending order by the relative accessibility score of each place for the given user, and top-n POIs are recommended.

3.4.1 Accuracy of Accessibility Based Recommendation

As described above, the top-n recommendation process is based on the sorted relative accessibility score of places. The relative accessibility of places depends on:

- Accessibility Rating $R(p, c_n)$ obtained using rating aggregation algorithm
- User's accessibility preference $P(u, c_n)$ obtained from user's profile

Here, $P(u, c_n)$ is the input from a user and is assumed to be accurate for each user. So, the accuracy of relative accessibility score $S_r(p,u)$ depends on the accuracy of the accessibility rating obtained using the user rating stream at that time. If $R(p, c_n)$ is closer to the actual state of accessibility of criteria c_n at POI p at the time recommendation is made, user's needs could be reflected in the recommendation.

CHAPTER 4:
EXPERIMENTS AND RESULTS

4.1 Simulation of Experimental Data

Biased coin flip was used to simulate the rating data for the study. The bias/probability would be changed within the period of time to reflect the change of situation of the place. At each step, the probability of getting 1 is p and getting 0 is (1-p). The value of a biased coin flip represents the ratings received by an accessibility category that has p as the confidence probability. Synthetic data is used because it allows us to test the aggregated value with the base confidence value used for rating generation at that point of time.

4.1.1 Pseudocode to generate synthetic data:

<u>Functions from Library:</u>

randomBoolean(chancesOfOne) : Generates a biased random Boolean value

getRandom(lowerBound,upperBound): Generates a random number within the bound.

<u>Pseudocode to generate Ratings:</u>

```
GenerateRandomRatings(noOfRatings,variation):
        currentConfidence=getRandom(0,100)
        ratings=[ ]
        confidence=[ ]
        for(i=0;i<noOfRatings;i++){
                if(randomBoolean(variation)){
                        currentConfidence=getRandom(0,100)
                }
                thisRating=randomBoolean(currentConfidence)
                ratings.append(thisRating)
                confidence.append(currentConfidence)
        }
        return [ratings,confidence]
```

The pseudocode accepts two input parameters. The first is an integer that decides the number of ratings to be generated. The second parameter is the variation probability represented by a positive floating point percentage value. The variation probability determines the chance that the probability of getting a positive rating (1) would be changed after generating each rating. Functions from Faker Library (Zaninotto, 2018) have been used to generate random Boolean value as well as a random integer value from a range.

4.2 Experimental Setup

Four series of experimental data were simulated to test the accuracy of the algorithm at different scales. The following table illustrates the size, nature and parameters used to generate the experimental data using the algorithm above:

Table 8: Size and nature of experimental data sets

Set #	Number of Ratings	Variation probability	Number of variation
1	100	1%	1
2	4,000	0.2%	3
3	100,000	0.2%	82

Number of ratings represents the count of ratings generated in the series. Variation probability is the probability parameter used to simulate the ratings using the algorithm above. Number of variations represents the number of times actual probability of getting a positive rating changed during the series.

First set of simulated data as shown in figure 15 consisted of 100 ratings. Initially, the probability of getting a positive rating was 10% which changed to 60% after 32 ratings.

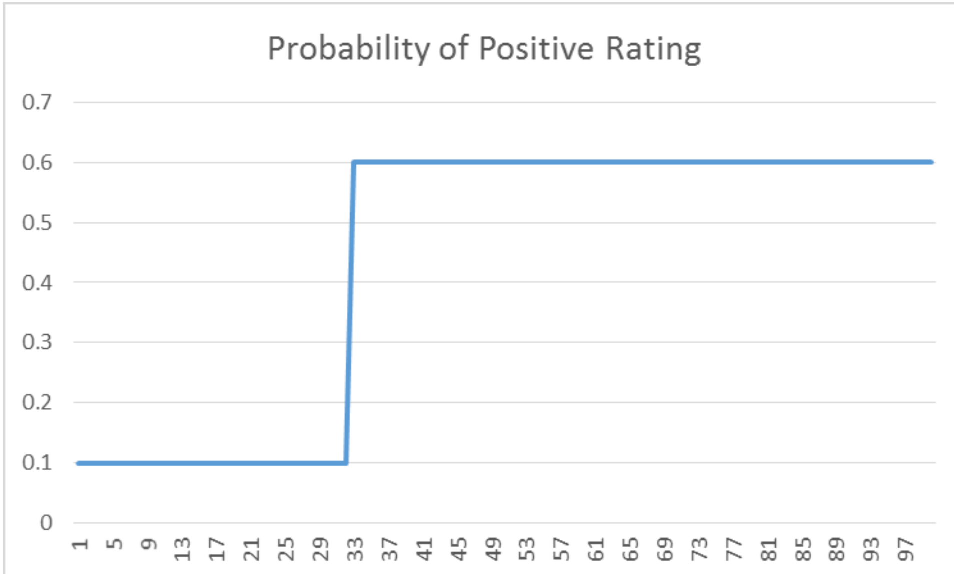

Figure 15: Probability of positive rating on simulated data (set 1)

The second series of simulated ratings contains 4000 ratings simulated using the variation probability of 0.2%. The variation of probability is demonstrated in figure 16.

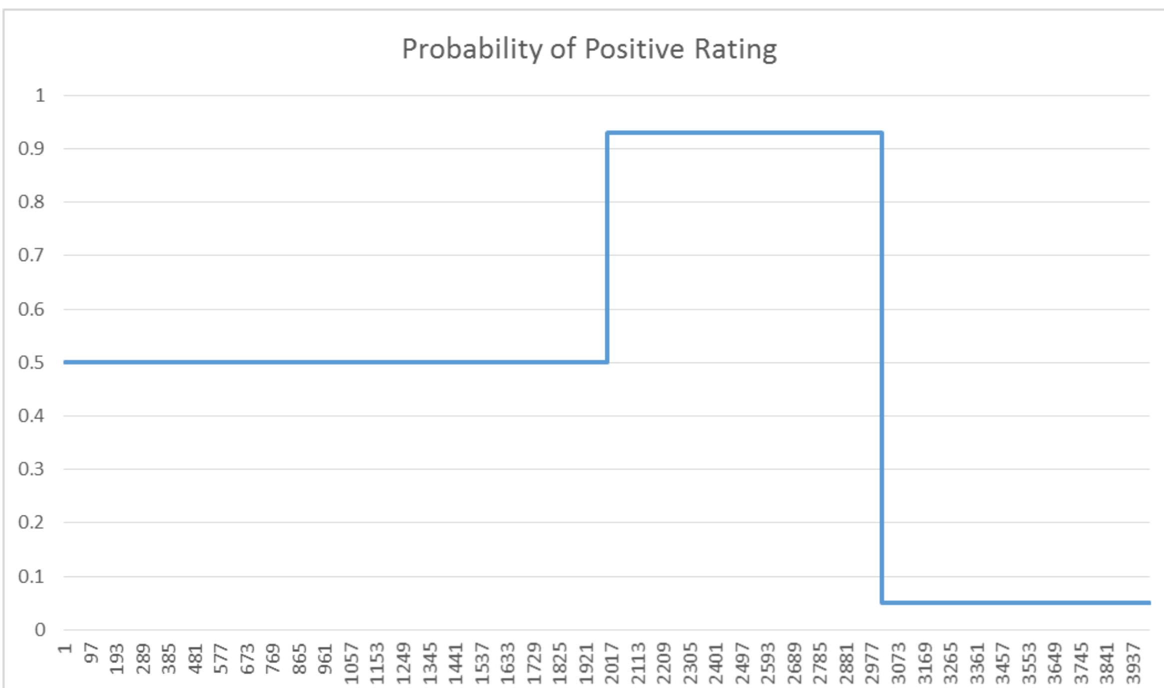

Figure 16: Probability of positive rating on simulated data (set 2)

The third series of simulated ratings contains 100,000 ratings simulated using the variation probability of 0.2%. The probability of getting a positive ratings changed 82 times throughout the series of series. Figure 17 shows the variation of probability of positive rating.

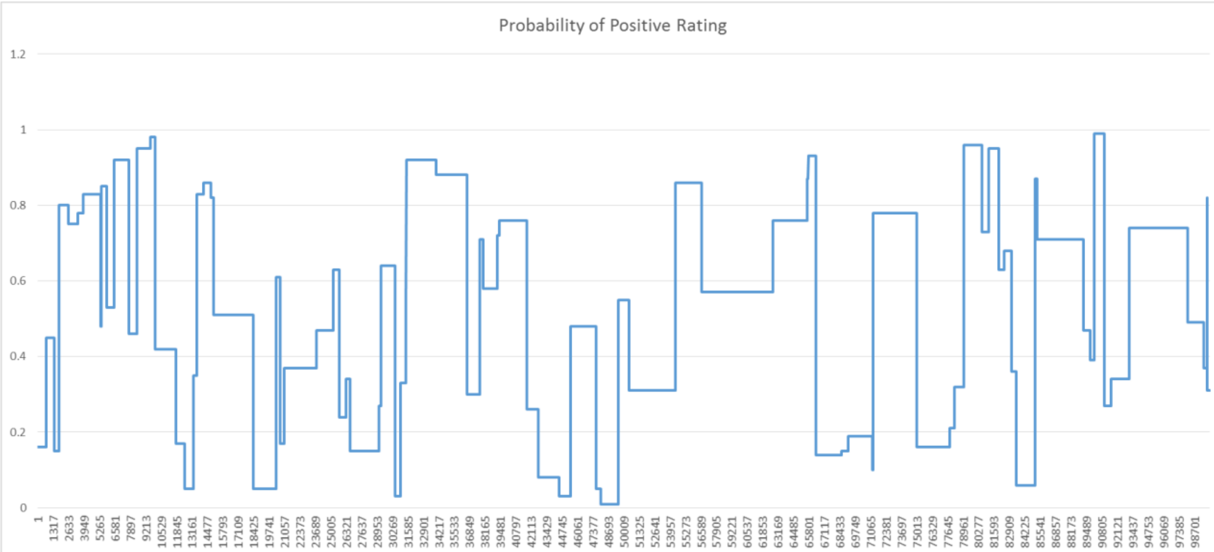

Figure 17: Probability of positive rating on simulated data (set 3)

4.3 Result and discussion

4.3.1 Evaluation Metrics

Accuracy:

Prediction accuracy of the system is calculated by comparing the predicted rating from the system with the real rating from the user. Mean Square Error (MSE) and Root Mean Square Error (RMSE), Mean Absolute Error (MAE) are most popular metrics used to evaluate the accuracy of the system. Given a test set T of user-item pairs $(\boldsymbol{u}, \boldsymbol{i})$ for which the rating $\boldsymbol{r_{ui}}$ by user u for item i are known, computed rating $\hat{\boldsymbol{r}}_{\boldsymbol{ui}}$ by user u for item i is calculated using the algorithm:

$$MSE = \frac{1}{|\mathrm{T}|}\sum\nolimits_{(u,i)\in T}(\hat{r}_{ui} - r_{ui})^2$$

$$RMSE = \sqrt{\frac{1}{|\mathrm{T}|}\sum\nolimits_{(u,i)\in T}(\hat{r}_{ui} - r_{ui})^2}$$

$$MAE = \frac{1}{|\mathrm{T}|}\sum_{(u,i)\in T} |\hat{r}_{ui} - r_{ui}\ |$$

Mean absolute error have been used for evaluation of aggregation algorithm. After each new rating, the aggregate is calculated and compared against the actual probability used to generate the rating. Since the actual probability changes across the series, the generated rating has to adapt to the change to minimize the error.

4.3.2 Results

For each series of ratings generated, aggregate have been calculated using different methods. For each method, MAE is calculated across the series using each data set mentioned above. Following are the aggregation methods used:

a) **Average**: For a series of ratings given:

$$Average = \frac{sum\ of\ all\ ratings}{number\ of\ ratings}$$

This is the aggregate used generally by majority of online rating system. If the opinion of people remains constant, this method would return the probability closest to the actual probability of getting a positive rating. But, since the actual probability changes within the series, this method does not adapt to the changes.

b) **Windowed Average**: Since the opinions of people is change, considering all the ratings together would result in high deviation from the actual opinion. So, considering a window of last 'n' ratings at a time, helps adapt to the change of opinion and reduces the error. We found that too small, or too large size of window would increase the error. Graphs below illustrate the variation of error against various window size. For each set of data, the window size with lowest error is considered for benchmarking.

Figure 18 illustrates the variation of error using different window size on test data set 2:

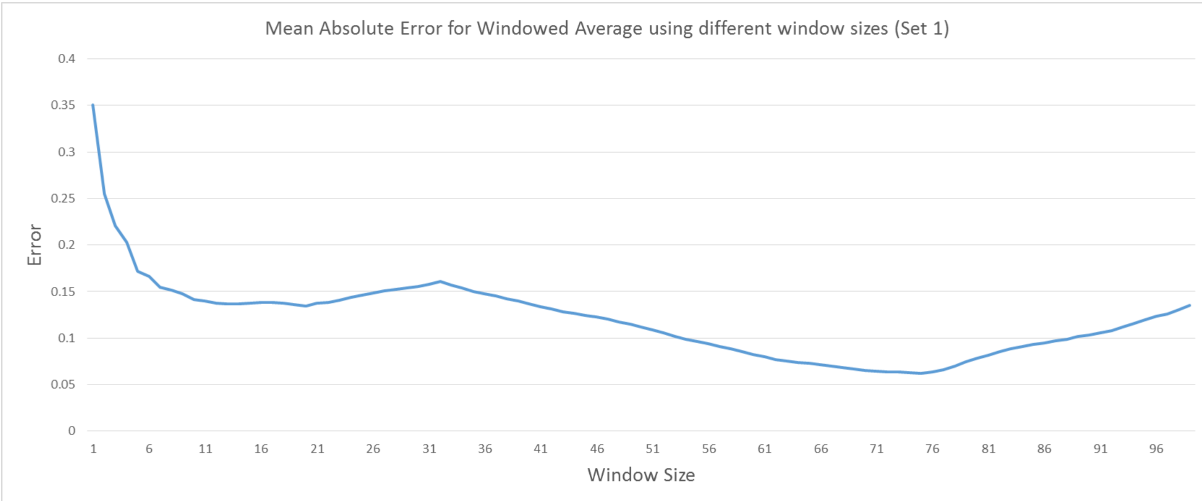

Figure 18: MAE for Windowed Average using different window sizes (Set 1)

For the first set with a series of 100 ratings and one variation, minimum MAE of 0.0624 was achieved for the window size of 75.

Figure 19 shows the variation of error using different window size on test data set 2:

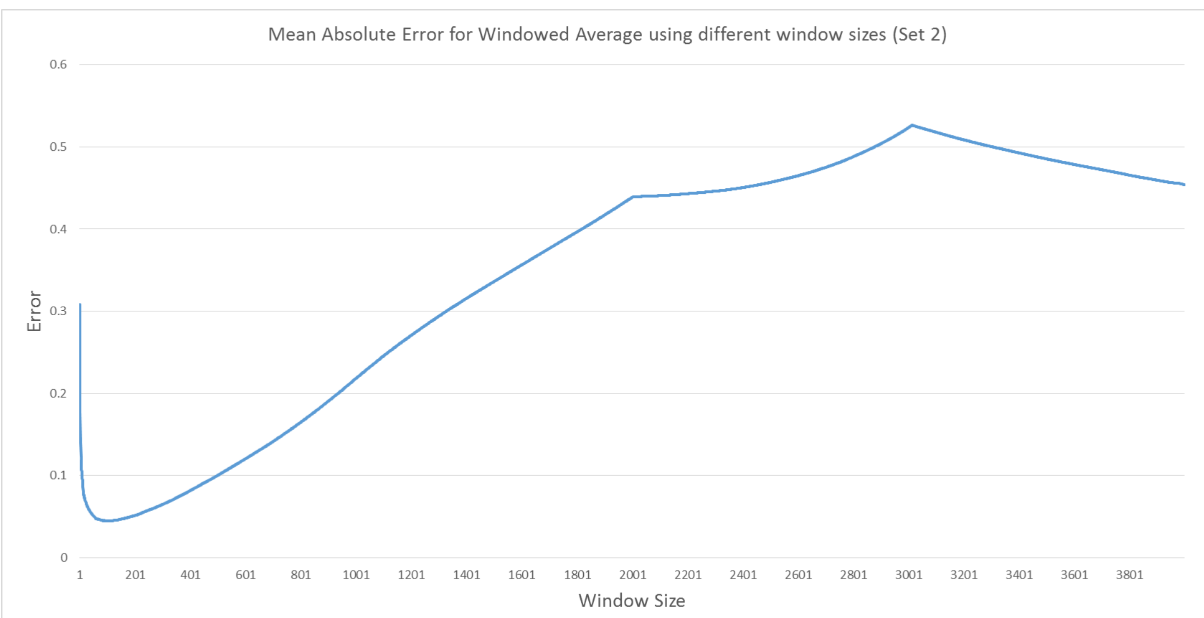

Figure 19: MAE for Windowed Average using different window sizes (Set 2)

For the second set with a series of 4,000 ratings and two variations, minimum MAE of 0.0445 was achieved for the window size of 100.

Figure 20 demonstrates the variation of error using different window size on set 3:

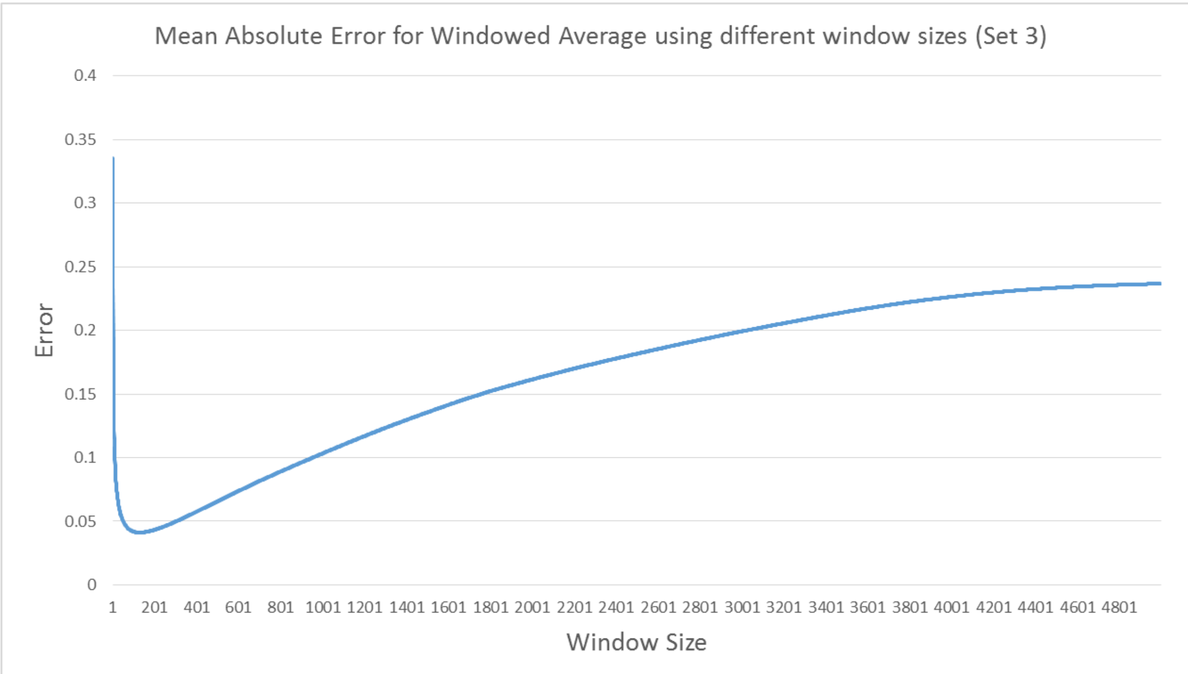

Figure 20: MAE for Windowed Average using different window sizes (Set 3)

For the third set with a series of 1, 00, 000 ratings and 82 variations, minimum MAE of 0.0410 was achieved for the window size of 127.

c) Voting: Voting is an aggregation technique in which the rating with highest frequency is considered to be the rating that represents the opinion of people. As regular average, this technique suffers due to lack of adaptation to change of opinion.

d) Windowed Voting: Similar to windowed average, windowed voting considers last 'n' ratings received in the series and selects the one with highest frequency in the window as the representative of opinion. Window size with lowest aggregation error is selected for each experimental data set.

Each of these aggregation techniques along with proposed damped aggregation was applied on each data set. The graph on figure 21 illustrates how the real probability is related to different aggregates computed using experimental data set 2:

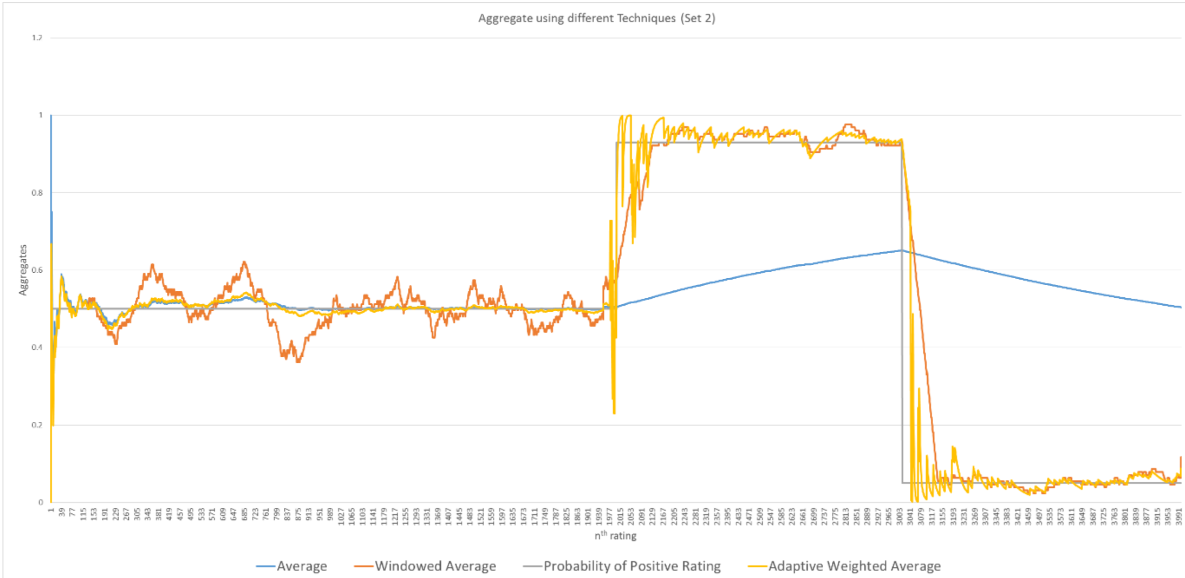

Figure 21: Relation between the real probability and aggregates (Set 2)

The following table summarizes the Mean Absolute Error (MAE) for each experimental data set using above and proposed aggregation techniques.

Table 9: Error for different Aggregation Technique

	Set 1	Set 2	Set 3
Average	0.194746091	0.21933805	0.25208762
Windowed Average	0.0624	0.04456154	0.04109851
Voting	0.44	0.5024625	0.4818915
Windowed Voting	0.448	0.28629487	0.25980545
Adaptive Damped Aggregate	0.175219153	0.02292323	0.03529679

Figure 22 illustrates the errors mentioned in the table above:

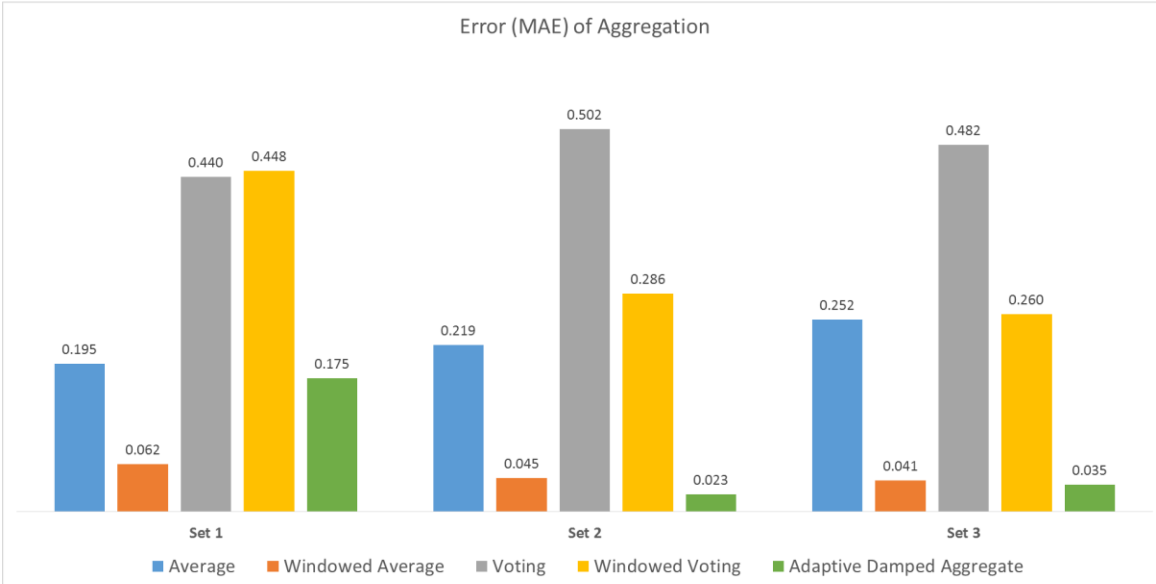

Figure 22: Error for different Aggregation Technique

4.3.2.1 Variation of accuracy with Sample Size

We simulated a set of 100 rating streams with different number of ratings on each to observe the change of error with the change of sample size. To identify the best window size, we iterated through all possible window sizes, calculated windowed average along the stream and selected the size with minimum MAE. In addition, we calculated non windowed average, voting, and windowed voting aggregate using the same window size. Then, we applied tunnel-based adaptive aggregation method on the generated ratings using the standard (z) score of 90% confidence and window size of 25. The following graph shows the variation of errors using different approach across different sample sizes:

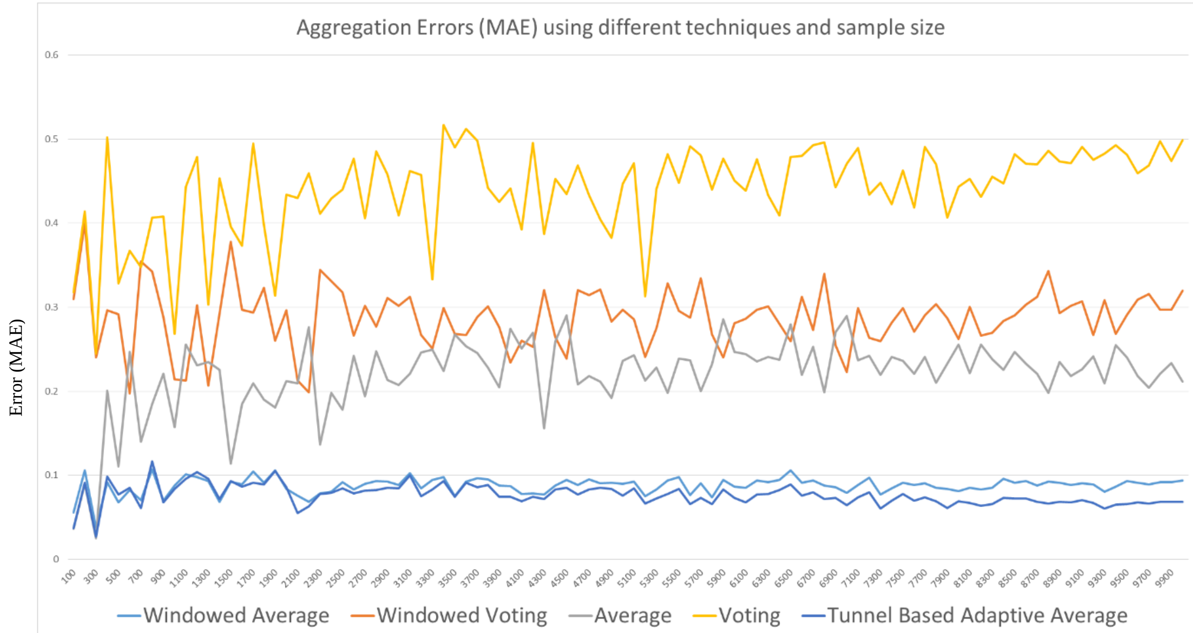

Figure 23: Aggregation Errors (MAE) using different techniques and sample size

The above experiment shows that the tunnel-based adaptive aggregate is more accurate for a larger sample size. The possible reason for this is that when the number of samples increases, the number of variations in probability within the stream also increases. Since, the tunnel-based adaptive aggregation adapts well to the change, the accuracy of this approach is better than others for higher sample size.

CHAPTER 5:
CONCLUSION AND FUTURE WORK

5.1 Conclusion

This thesis explores the problem of recommending places to disabled people based on accessibility ratings ant their disability profile. We propose a tunnel-based method to find the adaptive aggregate of binary time series ratings. This aggregate accurately represents the accessibility status of different accessibility criteria of the place at the given time. As places and their accessibility status are subject to change with time, this ensures that the recommendation adapts to the change of accessibility status. We compared the accuracy of the approach with other popular temporal aggregation techniques with sequence of simulated data. Proposed approach can be used for recommending POIs to disabled people with multiple disability who need more than one accessibility feature to visit a place. Such recommendation algorithm combined with other popular recommendation algorithm that takes factors like interest, social influence, etc. into account, can help disabled people discover places they can enjoy independently.

In addition to predicting the current state of accessibility at the place, the aggregation method proposed in this work can also be used for aggregating the goodness of items using implicit feedback in e-commerce portals. For example, by collecting the positive and negative events performed by users for a movie review, we can detect the current perspective of users towards that movie. As this can represent the current trend, the aggregated value can aid in generating better recommendation to users.

5.2 Future Work

With collaboration with organizations working with disabled people, and volunteers, real data about places and ratings can be collected. This can help validate the process and adjust the parameters that would allow accurate aggregation of ratings and hence help produce accurate and useful recommendation based on the accessibility ratings and the accessibility profile of the users.

This thesis discusses the use of explicit feedback from user for identifying current situation and recommending POI to users. Techniques for collecting implicit feedback can be studied and such feedback can be used to improve the accuracy of recommendation of places for people with disabilities. Data such as disability profile, time spent by the user at the place, frequency of visit, etc. can be utilized to implicitly decide the accessibility of the place.

In addition, this work makes an assumption that each user is trustworthy and considers each ratings received with equal importance. But, in order to portray positive or negative image of an organization, fake reviews might be added by users. So, using techniques that can predict the trustworthiness of users, we can reduce the impact of such fake reviews on the aggregate value calculated. This would further improve the trustworthiness of the recommendation generated.

REFERENCES

Access Locator. (2017). *Access Locator. [Web Application]*. Retrieved from http://accesslocator.com/

Access Now. (2017). *Access Now [Mobile Application]*. Retrieved from https://accessnow.me

Aggarwal, C. (2016). Model-based collaborative filtering. In *Recommender Systems* (pp. 71-136). Springer.

Ali, S., El Desouky, A., & Saleh, A. (2016). A New Profile Learning Model for Recommendation System based on Machine Learning Technique. *Indonesian Journal of Electrical Engineering and Informatics, 4*(1), 81-92.

AXS Map. (2015). *AXS Map. [Web Application]*. Retrieved from https://www.axsmap.com/

Baral, R., Zhu, X., Iyengar, S., & Li, T. (2018). ReEL: Review Aware Explanation of Location Recommendation. *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization* (pp. 23-32). ACM.

Bigham, J., & Ladner, R. (2011). What the disability community can teach us about interactive crowdsourcing. *interactions , 18*(4), 78-81.

Brabham, D., Ribisl, K., Kirchne, T., & Bernhardt, J. (2014). Crowdsourcing applications for public health. *American journal of preventive medicine, 46*(2), 179-187.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction, 12*(4), 331-370.

Chen, J., Li, X., Cheung, W., & Li, K. (2016). Effective successive POI recommendation inferred with individual behavior and group preference. *Neurocomputing, 210*, 174-184.

Cheng, C., Yang, H., Lyu, M., & King, I. (2013). Where You Like to Go Next: Successive Point-of-Interest Recommendation. *IJCAI, 13*, 2605-2611.

Chintagunta, P. K., Gopinath, S., & Venkataraman, S. (2010). The effects of online user reviews on movie box office performance: Accounting for sequential rollout and aggregation across local markets. *Marketing Science*, 944--957.

Ding, Y., & Li, X. (2005). Time weight collaborative filtering. In *Proceedings of the 14th ACM international conference on Information and knowledge management* (pp. 485-492). ACM.

Gao, H., Tang, J., Xia, H., & Liu, H. (2015). Content-Aware Point of Interest Recommendation on Location-Based Social Networks. In *AAAI* (pp. 1721-1727). AAAI.

Golab, L., & Tamer, Ö. (2003). Issues in data stream management. *ACM Sigmod Record, 32.2*, 5-14.

Gorawski, M., Gorawska, A., & Pasterak, K. (2017). The TUBE algorithm: Discovering trends in time series for the early detection of fuel leaks from underground storage tanks. *Expert Systems with Applications, 90*, 356-373.

Government of Ontario. (2005). *Accessibility for Ontarians with Disabilities Act S.O. 2005, c. 11*.

He, J., Li, X., Liao, L., Song, D., & Cheung, W. K. (2016). Inferring a Personalized Next Point-of-Interest. *AAAI.*

Herlocker, J., Konstan, J., Terveen, L., & Riedl, J. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS), 22*(1), 5-53.

Imrie, R. (2005). *Accessible housing: quality, disability and design.* Routledge.

James, B., & Lanning, S. (2007). The netflix prize. *Proceedings of KDD cup and workshop.*

Josang, A., Ismail, R., & Boyd, C. (2007). A survey of trust and reputation systems for online service provision. *Decision support systems*, 618--644.

Linden, G., Smith, B., & York, J. (2003). Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, *7*(1), 76-80.

Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In *Recommender systems handbook* (pp. 73-105). Springer.

Lyu, S. O. (2017). Which accessible travel products are people with disabilities willing to pay more? A choice experiment. *Tourism Management*, 404-412.

Megen, B., Grummon, C., Lobben, A., Omri, M., & Perdue, N. (2017). Accessibility in Pedestrian Routing. *Springer* (p. International Cartographic Conference). Cham: 479-492.

Mobasheri, A., Deister, J., & Dieterich, H. (2017). Wheelmap: the wheelchair accessibility crowdsourcing platform. *Open Geospatial Data, Software and Standards*, 27.

Oard, D., & Kim, J. (1998). Implicit feedback for recommender systems. *Proceedings of 5th DELOS Workshop on Filtering and Collaborative Filtering, 83*, 31-36.

Pazzani, M., & Billsus, D. (2007). Content-based recommendation systems. In *The adaptive web* (pp. 325-341). Springer.

Pope, A. M., & Tarlov, A. R. (1991). Disability in America: Toward a national agenda for prevention. Washington, DC: National Academy Press.

Quan, Y., & Cong, G. (2013). Time-aware point-of-interest recommendation. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval* (pp. 363-372). ACM.

Resnick, P., & Hal R., V. (1997). Recommender systems. *Communications of the ACM, 40*(3), 56-58.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 286-186.

Ricci, F., Lior, R., & Bracha, S. (2011). Introduction to Recommender Systems Handbook. In *Recommender systems handbook* (pp. 1-35). Springer.

Salton, G. (1989). Automatic text processing: The transformation, analysis, and retrieval of. *Reading: Addison-Wesley*.

Santani, D., Njuguna, J., Bills, T., Bryant, A., Bryant, R., Ledgard, J., & Gatica-Perez, D. (2015). Communisense: Crowdsourcing road hazards in nairobi. *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services* (pp. 445-456). New York, NY: ACM.

Santos, F., Almeida, A., Martins, C., Gonçalves, R., & José, M. (2017). Using POI functionality and accessibility levels for delivering personalized tourism recommendations. *Computers, Environment and Urban Systems*.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th international conference on World Wide Web, 2001*, 285-295.

Schein, A., Popescul, A., Ungar, L., & Pennock, D. (2002). Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 253-260). ACM.

Shani, G., & Gunawardana, A. (2011). Evaluating recommendation systems. In *Recommender systems handbook* (pp. 257-297). Springer.

Song, Y., Hu, Z., Leng, X., Tian, H., Yang, K., & Ke, X. (2015). Friendship influence on mobile behavior of location based social network users. *Journal of Communications and Networks, 17*(2), 126-132.

Statistics Canada. (2012). *Canadian Survey on Disability.* Statistics Canada.

Stern, D. (2004). Latitude and longitude. *Web page, NASA, Goddard Space Flight Center, Greenbelt, Maryland, 17*.

Su, X., & Khoshgoftaar, T. (2009). A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence, 2009*, 1687-7470.

World Health Organization. (2011). *World report on disability.* Geneva: WHO Press.

Xie, M., Yin, H., Wang, H., Xu, F., Chen, W., & Wang, S. (2016). Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. *CIKM 2016*, 15-24.

Ye, M., Yin, P., Lee, W.-C., & Lee, D.-L. (2011). Exploiting geographical influence for collaborative point-of-interest recommendation. *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval* (pp. 325-334). ACM.

Zaninotto, F. (2018). *Faker*. Retrieved from GitHub: https://github.com/fzaninotto/Faker

Zhao, P., Zhu, H., Liu, Y., Li, Z., Xu, J., & Sheng, V. (2018). Where to Go Next: A Spatio-temporal LSTM model for Next POI Recommendation. *arXiv preprint arXiv:1806.06671*.

VITA AUCTORIS

NAME:	Bidur Subedi
PLACE OF BIRTH:	Nepal
YEAR OF BIRTH:	1991
EDUCATION:	-Gandaki College of Engineering and Science (Pokhara University), B. Eng., Nepal, 2015 - University of Windsor, M. Sc., Canada 2018