


8-2013

Traveltant: Social Interaction Based Personalized Recommendation System

Sultan Dawood Alfarhood
University of Arkansas, Fayetteville

Follow this and additional works at: <http://scholarworks.uark.edu/etd>

 Part of the [Artificial Intelligence and Robotics Commons](#), and the [Graphics and Human Computer Interfaces Commons](#)

Recommended Citation

Alfarhood, Sultan Dawood, "Traveltant: Social Interaction Based Personalized Recommendation System" (2013). *Theses and Dissertations*. 805.
<http://scholarworks.uark.edu/etd/805>

This Thesis is brought to you for free and open access by ScholarWorks@UARK. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of ScholarWorks@UARK. For more information, please contact scholar@uark.edu, ccmiddle@uark.edu.

TRAVELTANT: SOCIAL INTERACTION BASED
PERSONALIZED RECOMMENDATION SYSTEM

TRAVELTANT: SOCIAL INTERACTION BASED
PERSONALIZED RECOMMENDATION SYSTEM

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Computer Science

By

Sultan Alfarhood
King Saud University
Bachelor of Science in Computer Science, 2007

August 2013
University of Arkansas

This thesis is approved for recommendation to the Graduate Council.

Dr. Nilanjan Banerjee
Thesis Director

Dr. Susan Gauch
Committee Member

Dr. Craig Thompson
Committee Member

ABSTRACT

Trip planning is a time consuming task that most people do before going to any destination. *Traveltant* is an intelligent system that analyzes a user's social network and suggests a complete trip plan detailed for every single day based on the user's interests extracted from the social network. *Traveltant* also considers the interests of friends the user interacts with most by building a ranked friends list of interactivity, and then uses the interests of those people in this list to enrich the recommendation results. *Traveltant* provides a smooth user interface through a Windows Phone 7 application while doing most of the work in a backend cloud service. To evaluate the results of the system, volunteers have rated the personalized results better than those results from only common factors such popularity and rating.

ACKNOWLEDGMENTS

I would like to express the deepest appreciation to my committee chair Dr. Nilanjan Banerjee, who has the attitude and the substance of a genius: he continually and convincingly conveyed a spirit of adventure in regard to research and an excitement in regard to teaching. Without his guidance and persistent help, this thesis would not have been possible.

I would like to thank my committee members, Dr. Susan Gauch and Dr. Craig Thompson, for the useful comments, remarks and engagement through the learning process of this thesis.

In Addition, special thanks are due to the staff of the University of Arkansas Graduate School for all of their help with theses. It would be difficult to make it through the semester without their help.

Also, special thanks go out to the faculty and staff at the University of Arkansas for their commitment to the University and to the students.

DEDICATION

To my parents, Dawood and Haya.

TABLE OF CONTENTS

1	INTRODUCTION	1
2	BACKGROUND	4
2.1	KEY CONCEPTS	4
2.1.1	PERSONALIZED SEARCH	4
2.1.2	PERSONALIZED SOCIAL SEARCH.....	5
2.1.3	SOCIAL NETWORK BASED PERSONALIZED SOCIAL SEARCH.....	5
2.2	RELATED WORK	6
3	SYSTEM DESIGN.....	9
3.1	DESIGN.....	9
3.1.1	FRIENDS INTERACTIVITY LIST	9
3.1.2	DISPOSABLE USER PROFILE.....	10
3.1.3	SEARCH CRITERIA	11
3.1.4	RESULTS CALCULATION.....	11
3.2	IMPLEMENTATION.....	13
3.2.1	TRAVELTANT CLIENT	16
3.2.2	CLOUD SERVICE	19
4	EVALUATION	25
4.1	MICRO BENCHMARKS.....	25
4.1.1	TRAVELTANT CLIENT.....	25
4.1.2	CLOUD SERVICE	26
4.2	MACRO BENCHMARKS	26
4.2.1	EVALUATION METHODOLOGY	26
4.2.2	EXPERIMENT	27
5	CONCLUSIONS	30
5.1	SUMMARY.....	30
5.2	FUTURE WORK.....	30
6	REFERENCES	32

LIST OF FIGURES

Figure 1: Friends' Interactivity Distribution Sample Graph	10
Figure 2: Traveltant's Sequence Diagram	14
Figure 3: Traveltant's Components	14
Figure 4: Traveltant's Flow Chart	15
Figure 5: Facebook Login Page Window	16
Figure 6: Destination Page Window	17
Figure 7: Trip Information Window	18
Figure 8: Trip Plan Window	19
Figure 9: Database ER Diagram	20
Figure 10: Phones used for results evaluation	28
Figure 11: Activity Rating Window	28
Figure 12: Activities rating average for both personalized and non-personalized versions.	28
Figure 13: Activities Yelp rating	29

1 INTRODUCTION

The travel industry is one of the most prominent industries nowadays, and many countries consider it as their major income source. Furthermore, thousands of companies are specialized in travel related services such as transportation, housing, food, and entertainment. Most of these services are provided through the internet whether it is in a direct way, such as a company website, or in an indirect way such as partners' systems.

In order to plan a trip, people need to contact a travel advisor to make their reservations and to suggest a trip plan. Alternatively, they can perform these tasks independently by searching the web and other media to create an appropriate plan that fits their preferences. There are thousands of free travel guides on the internet for all the popular destinations around the world. Generally, there are two types of travel recommendations: generic and personalized [1]. The generic recommendation answers the following question: "I am going to San Francisco ... what are the most popular attractions there?" The personalized recommendation, on the other hand, answers this question: "I am going to San Francisco ... what are the attractions which I will like the most?" The second question is more challenging to answer since people's interests vary based on factors such as age, gender, ethnicity, and personal preference.

Travelers refer to many web services for their trip planning, and one of the most popular websites is TripAdvisor¹, which maintains a huge users' run database of travel-specific content such as hotel reviews, destination attractions, and best destination restaurants. Another popular service is Yelp², which has an enormous database of business reviews around the world. Yelp is

¹ www.tripadvisor.com

² www.yelp.com

used by millions of people who review all types of businesses ranging from restaurants to medical clinics, and these businesses are categorized in more than 850 categories³.

Social networks have been playing an important role in connecting people in recent years, and they have been expanding tremendously by the number of users. For example, Facebook has more than 1.06 billion active users around the world, while Google+ has more than half a billion users in 2012 [2] [3]. People can share different types of information in these social networks from mentioning where they had lunch to posting photos from previous vacations. Moreover, people interact with each other in different ways including commenting on each other's activities and posting photos that they have taken together. All of this information can be analyzed to create a picture of people's connections and relationships. Consequently, this information can help to identify some of people's interests and preferences.

Using current technologies, a smart system can be designed to serve as a travel advisor for individuals. This system can suggest the proper plan for every user based on several criteria, such as the individual interests of the user and his similar friends', the traits of the user (gender, age, and other demographic data), the popularity of the activity suggested, and many others.

Traveltant is an intelligent system that analyzes a user's Facebook account and suggests a complete trip plan detailed for every single day based on the user's interests mined from the user's social network. *Traveltant* also considers the interests of friends the user interacts with most by building a ranked interactivity list, and then uses the interests of those friends to enrich the recommendation results. *Traveltant* is integrated with several web services including Yelp [4] to retrieve destinations' popular attractions and Bing [5] to validate and geo locate

³ http://www.yelp.com/developers/documentation/all_category_list

destinations. Additionally, *Traveltant* is implemented as a backend cloud service with a smooth user interface through a Windows Phone 7 application.

The remainder of this document is structured as follows. In Section 2, background information is introduced to explain the concept of personalized search, and afterward, several related works are discussed. Following, the system design is illustrated in section 3, which contains both the abstract design of the system and the implementation details. In section 4, the system is evaluated in different levels, and the results of the social network based personalization are compared with non-personalized results. Finally, a conclusion with a discussion on future work is presented in section 5.

2 BACKGROUND

2.1 Key Concepts

2.1.1 Personalized Search

Contextual computing refers to “the enhancement of a user's interactions by understanding the user, the context, and the applications and information being used, typically across a wide set of user goals” [6]. This concept is not just about considering individuals’ preferences, but it is a wider concept that includes adapting information systems for each user and each point of computation [6]. Personalized search is a part of the contextual computing concept that focuses on utilizing information about specific users to adapt search results in order to meet those users’ interests without explicit users’ input. Pitkow et al. [6] define two different approaches of personalized search; the first is to re-rank search results, and the other is to adjust users’ queries to match their profiles. In general, the user’s interests are gathered by different approaches including the user’s previous history [7], the user’s interaction with the system such as mouse clicks and eye movement [8], and the user’s similar and related people’s interests [8] [9] [10]. These user’s interests are represented and stored as a user profile, which can be used whenever a personalization is required. There are many difficulties using the user profile, and the most significant one is the fact that many people consider saving it as a privacy violation. This difficulty can be overcome by designing the user profile to contain only the minimum information required to achieve the personalization. Moreover, information masking can be used to conceal people’s private information into categories instead of saving it as a whole. For instance, a system that requires the list of businesses visited in a year can be designed to store only the category of these businesses, and if their locations are essential, they can be stored as approximated areas instead of accurate geographical coordinates. Another suggested solution to

the privacy concerns is to avoid saving the user profile. In this way, the user profile is generated every time the personalization is needed, and it is deleted once used. Still, the disadvantage of this approach is the decrease in system efficiency since the user profile is generated whenever personalization is required.

2.1.2 Personalized Social Search

Despite the issue of privacy concerns, in recent years Web 2 applications such as blogs, Wikis, and social network sites have spread, and people have become more willing to share information with the public. These social services host several types of content including images, documents, posts, and people's connections, and with these social services, search can be personalized based on people's published content [11]. A user profile can be constructed to model the user's interests from his or her social service published content. For example, a user who publishes frequent articles about information technology can be assumed to be interested in this field and his queries can be personalized to reflect these interests. The advantage of using publicly published content is to avoid the privacy issues since any user willing to share such information would not mind using it for personalization [9]. Additionally, by using profiles created from publicly published content, people can be grouped together based on their similar interests, an approach that can be used to expand users' profiles for further personalization.

2.1.3 Social Network Based Personalized Social Search

Modern social networks are not just designed to host publicly shared content, they go beyond this functionality by hosting different types of people's relationships and communications. In social networks like Facebook, people can specify different types of relationships with other people, and can interact with these people in different ways including commenting on and liking each other content such as photos and status updates. In addition,

people can specify with whom they want to share their content, and they can perform private communications such as messages and chatting without exposing these communications to the public. This fact makes people relaxed to carry on their social network interactions to the next level, and as a result, these social networks become a rich environment for personalization. Social network data can be analyzed to build rich users' profiles that reflect their interests more accurately. Furthermore, users' relations can be utilized to draw a better and more accurate picture of people's interests since people connect more with those people who share the same interests with them. The assumption behind this approach is that a user's interests are similar to those users who are considered related and similar to him or her. This approach is important especially when a user has limited social network activities, making the process of building the user profile harder. This project adapts this approach for personalization since it uses the user's interests and includes his most interactive friends' preferences upon recommendation of the trip plan.

2.2 Related Work

In recent years, researchers have been exploring different ways to personalize search results for users in order to provide relevant results for their needs. In the field of social networks, Carmel et al. [9] investigated personalized social search based on the user's social relations by using familiarity and similarity approaches. Their implementation uses the IBM Lotus Connections (IBM LC) platform for the enterprise to personalize users' search queries. Working on this limited and targeted social network makes it difficult to decide if the personalization results are good enough to generalize for real life social networks taking into consideration that the relationships in the IBM LC are limited to career related relationships. Similar to this approach, *Traveltant* utilizes social networks based relations to personalize

results; however, *Traveltant* uses the broader social network, Facebook, to get users' preferences based on both their interests and the interests of friends with whom they interact most.

Furthermore, *Traveltant* considers other common factors for personalization such as popularity and rating of the items suggested.

Another similar approach is suggested by Golbeck [12] to recommend movies for users based on their social network relations. This approach uses explicit trust relations where users choose other users who are relevant to, and then recommended movie titles are generated based on their cumulative preferences with those users. This method considers users' preferences based on the explicitly provided user's list, and it does not build implicitly based on the users' interaction as in *Traveltant*. Additionally, [13] [14], and [15] have explored social network relations for content personalization using different approaches. However, none of these works have considered grouping friends based on their closeness to the user, and whether these friends' interests should be considered for the user personalization process or not.

Using other social networks, [16], [17], and [18] have used picture collections from photos based social networks to mine travelers' activities in order to recommend the appropriate travel tips for people. Different from [16] and [17], [18] has not just mined photos from the web, they have considered specific user profiles for further personalization. They have handled photo attributes to obtain relevant information such as gender, age, and race of the photo owner, and then considered them to personalize trip suggestions. They have proposed a probabilistic recommendation approach based on the user's profile to recommend a suitable trip suggestion for him or her. However, expanding this approach to include more personalized methods including analyzing the user's connections will be highly effective to produce better suggestions, an approach that *Traveltant* adapts.

There are other works that use personalized search to recommend trip advices but do not utilize social networks. Crumpet [19] is a trip planner system, developed by European researchers, which uses the domain of user's interests to match the appropriate activity to every user. Besides, Crumpet learns users' interests based on their interaction with the system, and then personalizes trip recommendations; on the other hand, *Traveltant* uses the users' social network to learn their interests automatically and then provides a suggestion of a complete trip plan instead of providing one suggested activity as Crumpet does.

Finally, Murshid [20] is another mobile application that works as an automated tourist advisor. Murshid detects the current location and context of the traveler and guides him or her in a destination based on his or her location. Unlike Murshid, *Traveltant* suggests a personalized trip plan based on his or her social network analysis. In addition, *Traveltant* provides a complete and detailed trip plan for every day instead of providing a single suggestion as in Murshid.

3 SYSTEM DESIGN

3.1 Design

Personalized search utilizes users' preferences and interests as gathered from different sources to reevaluate and adopt search results [9]. *Traveltant* examines users' social networks to obtain such preferences based on users' interests as well as their most interacted friends' interests. Thus, personalized trip plans are produced in light of these preferences, which are to be represented as a user profile.

3.1.1 Friends Interactivity List

To include friends' preferences, *Traveltant* analyzes all the user's public activities in their social network to build an interactivity ranked list. This ranked list is used as an indicator to whether this friend's interests should be included in the user's profile or not. In other words, taking into account all the friends' interests will eventually be equal to the universal popularity and rating factors if the user has infinite number of friends. Each friend gains more scores whenever he or she interacts with the user in any public activity such as being in the same image, commenting on a user's post or photo, or writing a public post for the user. Consequently, those friends with high interactivity scores are considered more related to the user [10]. To illustrate, the friends' interactivity distribution graph in Figure 1 shows that there are 153 friends for a user X, and obviously, the user's relationship is not the same with everyone in his friends list. Some of these friends can be family members while some others are work related individuals; however, only 10 friends in his entire list have an interactivity score of 50 or more since they have engaged with the user X in many social network activities. Therefore, in addition to the user's preferences, only those 10 friends' interests will be taken into consideration upon personalizing the trip plan for the user X.

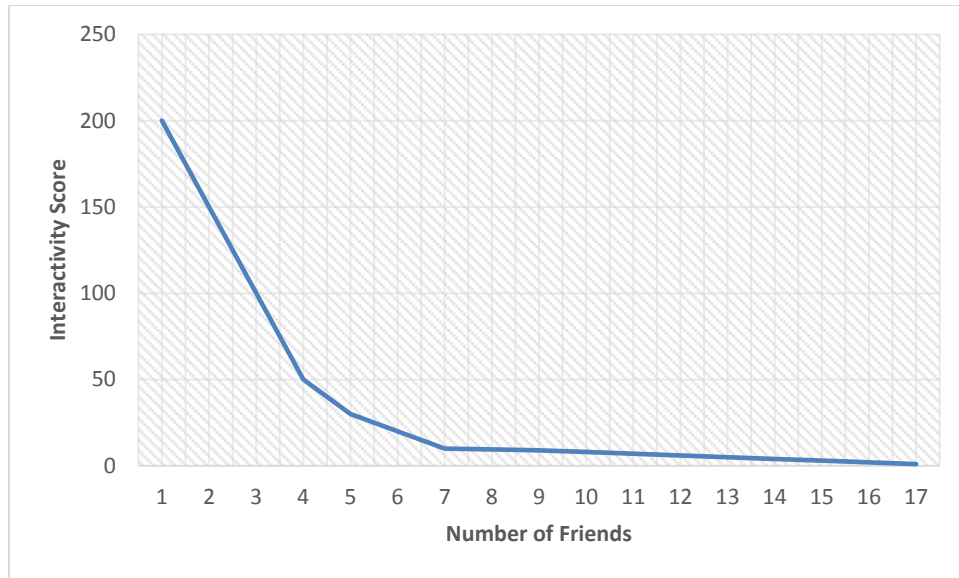


Figure 1: Friends' Interactivity Distribution Sample Graph

3.1.2 Disposable User Profile

As discussed previously, many people consider saving their users' profiles as a violation of their privacy and for this reason, *Traveltant* utilizes an approach we call the disposable user profile. In this approach, the user profile is generated every time the user accesses the system, and it is only valid for that particular session. Once the user terminates this session, the user profile is deleted. This approach can be achieved by the careful design of the user profile building process to maintain the system efficiency. Only essential user's information is gathered to build the user profile, and the selection process of this information is based on the particular social network specification. The detailed specification of the information used in this project is discussed in the implementation section of this document since it is specific to the Facebook environment.

3.1.3 Search Criteria

In order to produce a trip plan for a user, *Traveltant* considers many factors to rank and personalize results. Some of these factors are related to the user and others are related to the activity/business itself. These factors include:

- Popularity of the activity/business
- Rating of the activity/business
- Location of the activity/business
- Gender and age of the user (from the social network)
- User's interests (from the social network)
- User's most interacted friends' interests (from the social network)

3.1.4 Results Calculation

A user profile is generated to cover all the users' preferences from their social network as well as their most interactive friends. Utilizing this user profile, personalized results are calculated using the following method [9]:

$$R(L, P(u)) = \alpha \sum_{i \in R_{np}(L)} i + (1 - \alpha) \left[\beta \left[\sum_{i \in R_{np}(L)} \sum_{j \in C(u)} c(i) \cdot w(j, u) \right] + (1 - \beta) \left[\sum_{i \in R_{np}(L)} \sum_{f \in F(u)} \sum_{j \in C(f)} c(i) \cdot w(f, u) \cdot w(j, f) \right] \right]$$

Where:

- L stands for location
- $P(u)$ is the profile of the user u
- R_{np} is the non-personalized result generated by using only common factors like the popularity and rating of an activity
- $c(x)$ stands for the category of an activity x , and $C(u)$ is the collection of categories which the user prefers

- f stands for a friend, and $F(u)$ is the ranked list of the user's friends based on their interaction
- $w(x)$ stands for the weight of x

The final personalized results $R(L, P(u))$ includes the non-personalized results based on the popularity and location of the activity/business only, and the personalized results that contain the user's interests extracted from his or her social network. In this way, the recommendation results could contain a must-see-attraction that the user should try even though his profile does not refer to anything related to this activity. The amount of such activities can be adjusted by changing the value of α to specify how much personalization is required. Furthermore, personalized results consist of two parts: user's related results ($\sum_{i \in R_{np}(L)} \sum_{j \in C(u)} c(i) \cdot w(j, u)$), and his most interacted friends' related results ($\sum_{i \in R_{np}(L)} \sum_{f \in F(u)} \sum_{j \in C(f)} c(i) \cdot w(f, u) \cdot w(j, f)$). The first part is about re-ranking the non-personalized results based on the user's interests. Each activity is weighted based on the weights of the user's interests as mined from his social network account. Likewise, the friends' related results are re-ranked based on their interests as extracted from their social network account. Similar to the non-personalized results, the ratio of the user's interests to his most interacted friends' interests can be tuned by manipulating the variable β .

3.2 Implementation

In order to achieve an efficient and scalable architecture, *Traveltant* is designed as two separate components: a user interface and a backend cloud service. The user interface interacts with the user and takes advantage of his or her system environment's features such as GPS, which provides a smoother way to get the user's current location instead of typing it. A Windows Phone 7 application has been implemented as the user interface. Simultaneously, the backend cloud service performs all the core work of the system from analyzing the user's social network to generating the personalized trip plan.

Dividing the system into two parts relies on three basic reasons. The first one is to separate the core functionality from the user interface. In this way, various user interfaces can be implemented without the need to re-implement the core functionality. Secondly, users have the ability to interact with the system among multiple operating system platforms including mobile systems. Thirdly, this approach is highly efficient for mobile platforms since most of the processing is performed in the cloud part, a process that contributes to save energy consumption.

Several connections are going back and forth between the client and the cloud service in addition to the connected third party platforms as illustrated in the following sequence diagram (Figure 2):

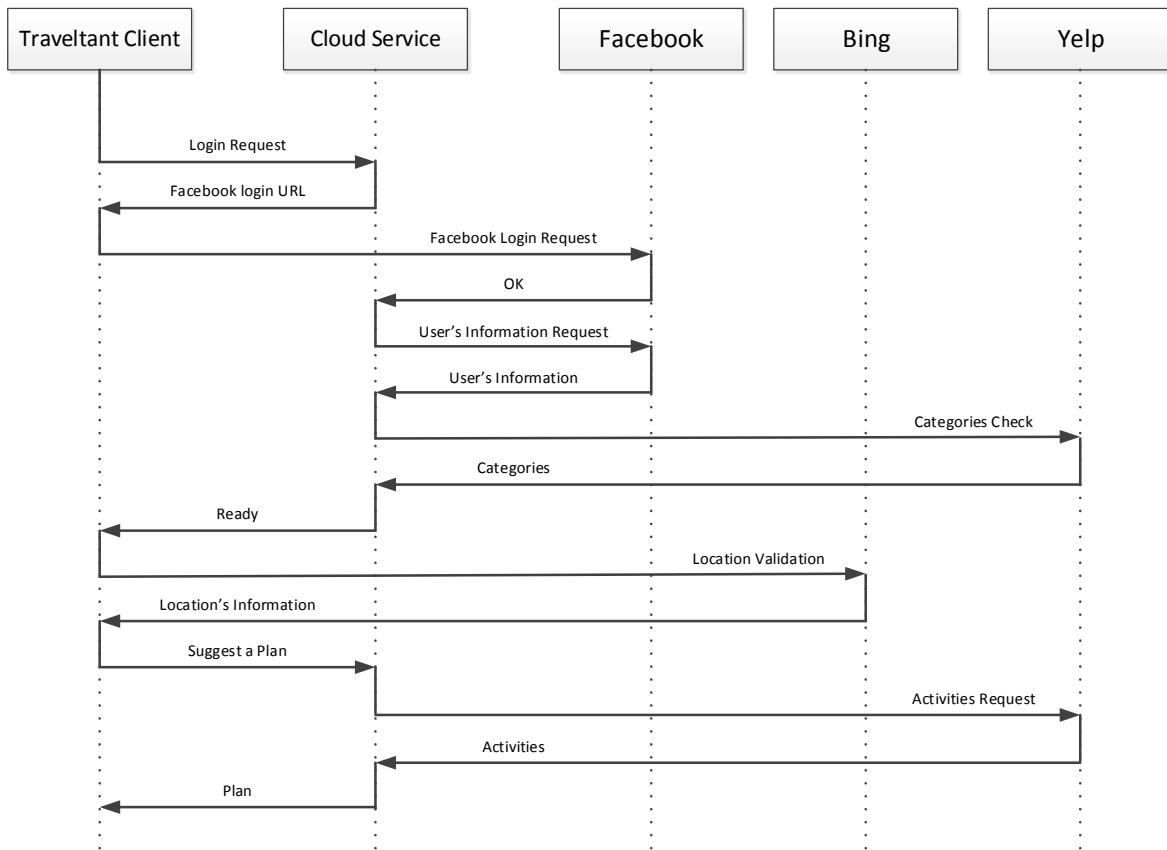


Figure 2: Traveltant's Sequence Diagram

Furthermore, to achieve better scalability and integration, both the client application and the cloud service are architected into several components as in the following diagram (Figure 3):

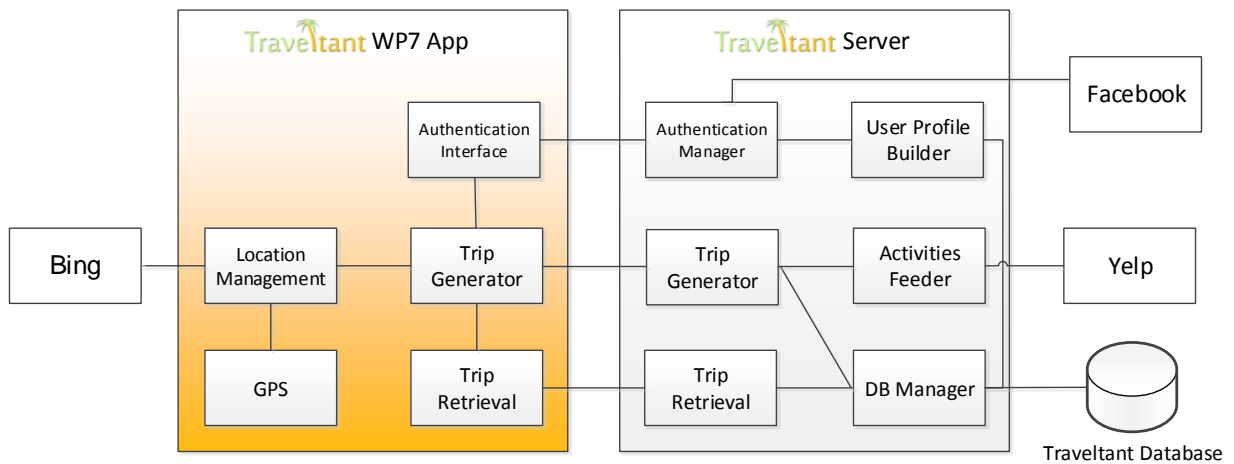


Figure 3: Traveltant's Components

The system has been built with a smooth user experience taken in respect. The user needs to supplement the basic information about his or her trip, and the system will automatically generate the personalized trip. The flowchart of the system for both client and cloud service is shown in Figure 4:

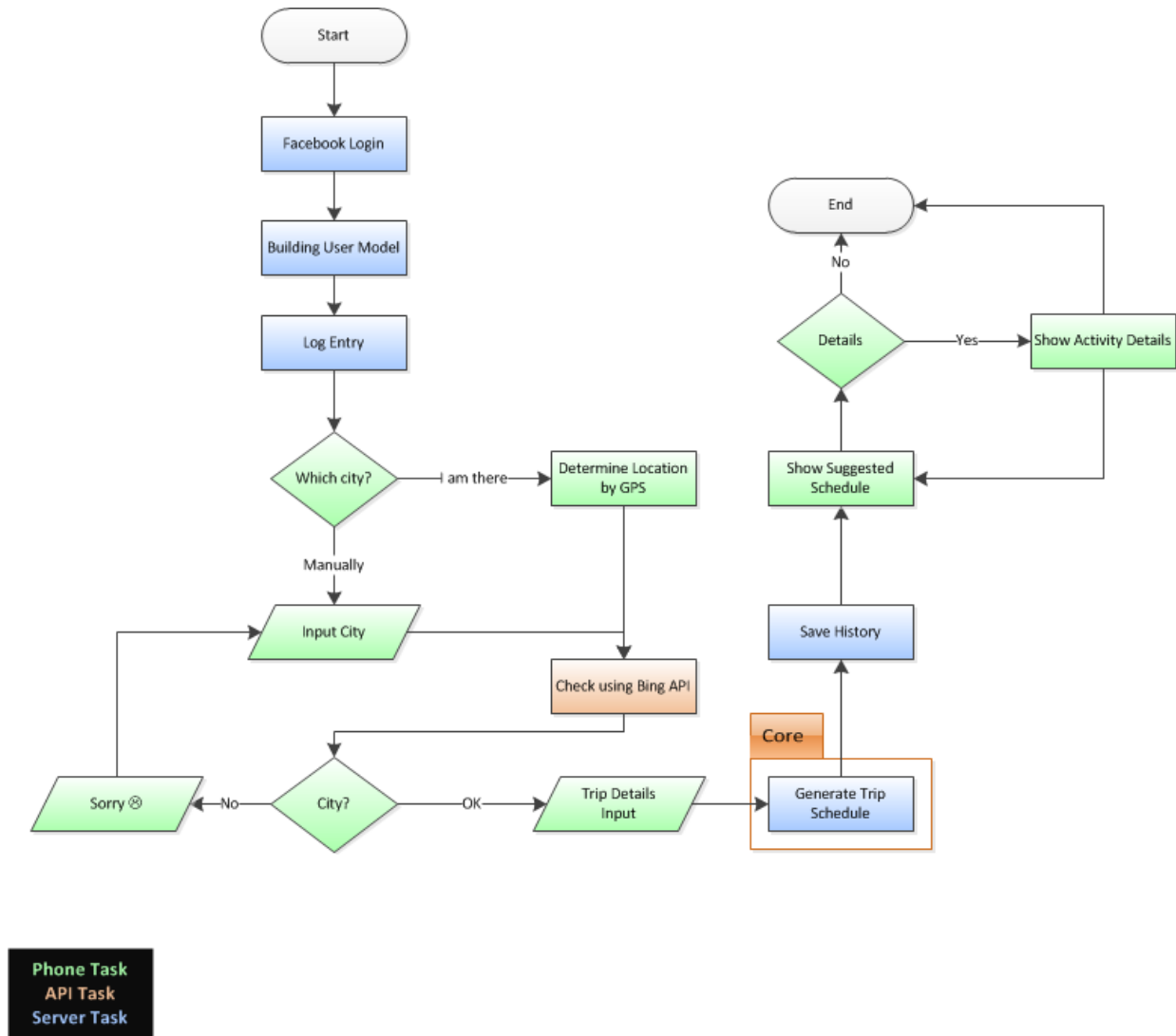


Figure 4: Traveltant’s Flow Chart

3.2.1 Traveltant Client

Traveltant client is implemented as a Windows Phone 7 application, using the Microsoft Visual Studio environment. Once the user opens the application, a request to the server is sent asking for the right Facebook login page as illustrated in Figure 2, and then this page is shown to the user in a web component as in Figure 5.

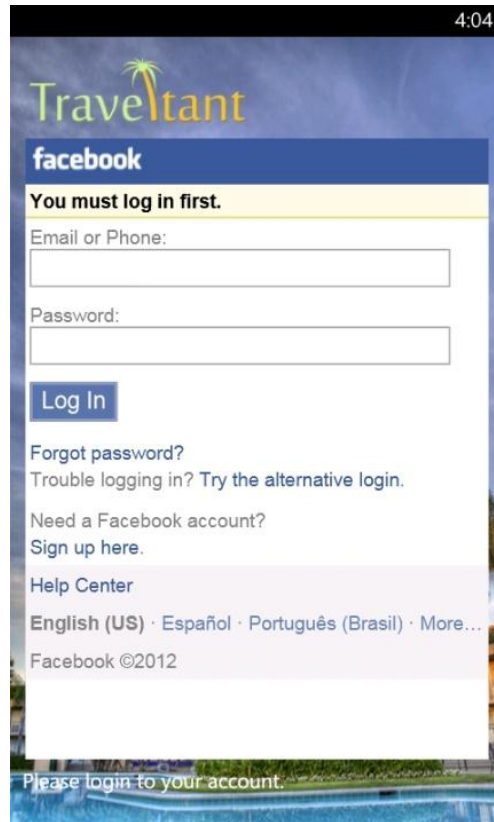


Figure 5: Facebook Login Page Window

The Facebook login process is handled through Facebook API [21] directly to maintain the privacy of the system users. To illustrate, *Traveltant* does not store or handle the users' Facebook credentials. Rather, these credentials are handled directly by Facebook, while *Traveltant* gets only a temporary access token, which can be used to query Facebook for the user's information as the user is logged in. Once the user logs out from Facebook, this access token becomes not

valid anymore to query the user's information. When the login process completes, the server starts building the user profile and sends the user identification number (ID) to the application, which opens the destination window as in Figure 6.

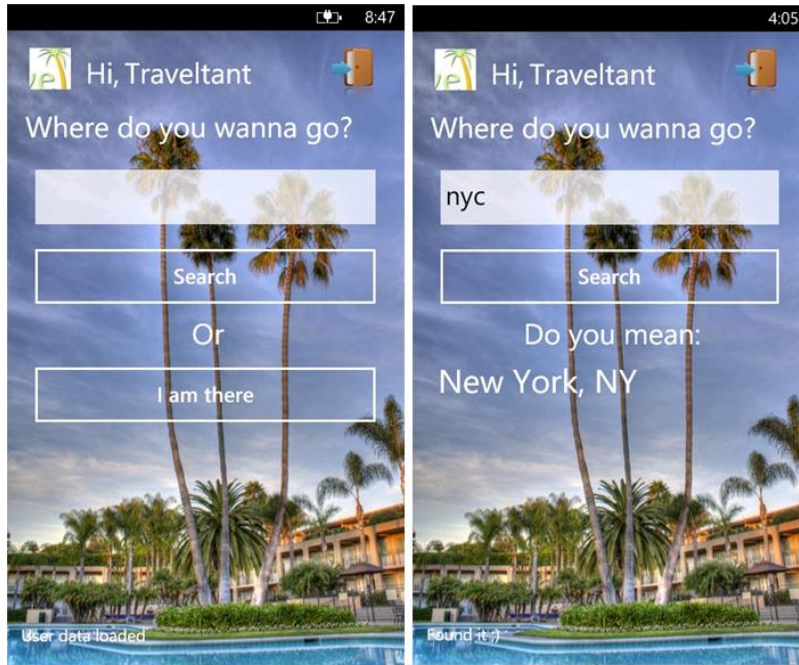


Figure 6: Destination Page Window

In the destination window, the user may type the destination or use the device GPS to detect his or her location. The application validates the user's input using Microsoft's Bing API [5] and confirms it. Besides, this API is used to geo-locate the destination to acquire its coordinates, which are used in the system as the main way to represent locations. Afterward, the trip details including the date and duration are inputted through a sleek interface as in Figure 7, and a request is sent to the server to generate a personalized trip plan with the provided details.



Figure 7: Trip Information Window

When the server completes generating the personalized results, it sends the detailed plan to the application, which displays it in the results window. The user can access each activity detail by touching it, and then a set of complete activity details is displayed including a picture, a phone number, and a map location as showed in Figure 8.



Figure 8: Trip Plan Window

3.2.2 Cloud Service

The cloud service is the main part of the system where most of the processing is performed, from creating users' profiles to generating personalized trips.

The cloud service is implemented in PHP 5 environment, and it is connected to a MySQL 5 database where the system keeps records. This database is designed as in the following ER diagram (Figure 9):

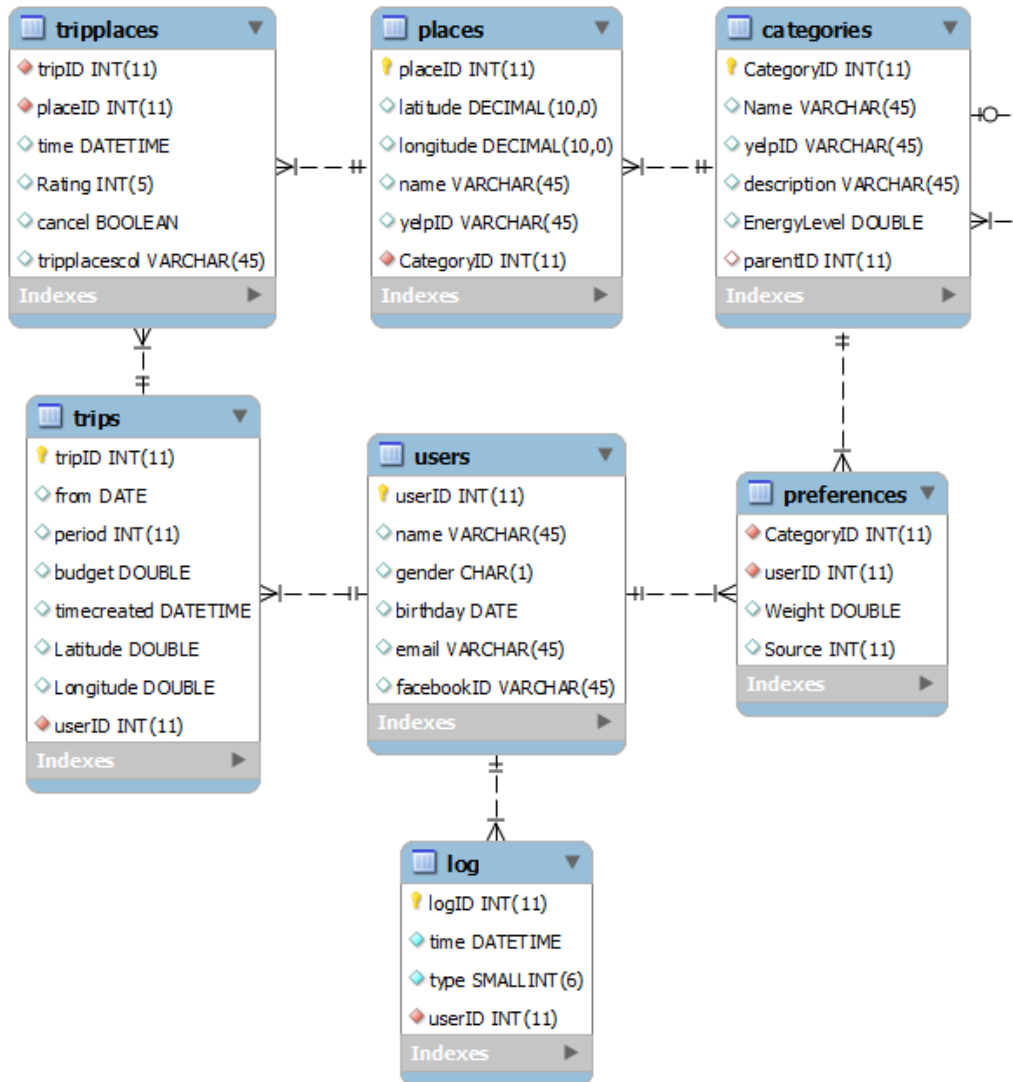


Figure 9: Database ER Diagram

Once the user logs in successfully into Facebook, the server receives an access token from the Facebook API, and this token is used whenever the server connects to the user’s Facebook account. Using the Facebook account’s access token, the server starts building the user’s profile.

Building User Profiles

At the beginning of the process of building a user profile, the server retrieves the friends list of the user and starts ranking them based on their interactivity scores. The interactivity score of every friend is calculated by counting the following actions:

- A friend comments on one or more of the user's:
 - Photo albums
 - Uploaded or tagged photos
 - Wall posts
 - Links
 - Notes
- A friend likes one or more of the user's:
 - Photo albums
 - Uploaded or tagged photos
 - Wall posts
 - Links
 - Notes
- A friend writes on the user's wall posts
- A friend tags the user in one or more of his or her posts

As discussed previously, these actions are selected carefully for performance issues since retrieving data from Facebook API is done through the Facebook FQL technology [21], which offers a limited and restricted way to query data from Facebook. Due to the massive data required to calculate an accurate interactivity score, a highly efficient way is designed to query all the previously mentioned data using only a few queries, a process which saves considerable bandwidth and increases response time while retaining the disposable user profile approach.

After creating the friends interactivity list, the server queries all the activities which the user has liked or checked in before, and likewise it performs another similar query for everyone

in his most interactive friends list. Currently, this operation is being limited to restaurants activities only due to Facebook API limitations for the public⁴.

A major issue of this queried list of restaurant activities is the lack of categories for every restaurant, information that Facebook does not offer; hence, it cannot be used for personalization. To illustrate, everyone favors different types of restaurants (such as Italian, Chinese, and Indian), and the lack of these categories makes it challenging for personalization. This issue is resolved by querying these results with the massive Yelp database [4] through matching businesses' names, phone numbers, and locations to get the corresponding category for every restaurant. As a result, a complete Facebook and Yelp integration layer is implemented to expedite this process. Finally, in addition to adding a log entry in the system's database, these categories are stored in the database along with its sources and weights to create the user profile.

Generating Personalized Plans

Upon receiving trip requests from the client containing desired destination and trip details, the server retrieves the user profile from the database. Moreover, it requests ranked lists of activities including restaurants using the Yelp API [4] with the provided coordination. These lists are ranked using the popularity factor in Yelp, and each entry contains different details including ranking and location information. Every requested list is related to a particular activity provided for the user including food activities, and these lists are queried using the following parameters in Yelp API:

⁴ Facebook limits the number of API queries for their general API users, and it requires a special agreement for their partners to allow them to increase the number of allowed queries.

Activity Type	Yelp Keywords	Keyword Description
Breakfast	breakfast_brunch	Breakfast & Brunch
Lunch and Dinner	Restaurants	Restaurants
Daylight Activity	active, arts, tours	Active Life, Arts & Entertainment, Tours
Night Life	Nightlife	Nightlife

The server assigns different weights for every activity retrieved, and then chooses the activities with the highest weights. The final weight is combined from the following sub weights:

Name	Scale	Use	Description
Location	1-10	All activities, except the first activity of every day	The distance of this activity from the first day activity of every day as the following: <ul style="list-style-type: none"> • Less than 1000 meters: 10 • Less than 2000 meters: 8 • Less than 3000 meters: 6 • Less than 4000 meters: 4 • More than 4000 meters: 2
Activity Rating	1-20	All activities	This weight is equal to the stars number in the Yelp rating multiplied by 4
Activity Popularity	1-20	All activities	This weight is based on the ranking of results using the Yelp popularity factor
User's Preference ⁵	1-25	Restaurants	This weight is related to the user's profile generated from his social network
User's Friends Preferences ²	1-25	Restaurants	This weight is related to the user friends' profile generated from their social network

Next, the server filters the results based on their Energy Level, which is a statically assigned value for every activity category. The goal of this Energy Level is to deploy the age factor of the user, so only those activities that fit the user are recommended. Currently, the Energy Level values are assigned as the following:

⁵ As previously mentioned, social network based personalization is currently limited to restaurants related activities which these factors applies to. Others are recommended using the other factors.

Age	Gender	Energy Level
15-30	Male, Female	A
31-45	Male, Female	B
46-60	Male, Female	C
<60	Male, Female	D

The concept of the Energy Level is important and can be further investigated based on two different approaches. The first approach is to study the optimal default values of every activity so it can be used for every recommendation request. Secondly, these values can be adopted based on the user nature, and its value can be integrated with the user profile. In this project, static values are selected to calculate the Energy Level, but dynamic values will be carried out for future work on this system.

Finally, the generated trip plan is compressed and sent to the client in an XML format that contains the following activity attributes:

Activity Attribute	Description
ID	<i>Traveltant</i> assigned activity ID
Suggested time	Which time <i>Traveltant</i> suggests this activity
Name	The name of the activity or the business
Description	The description of the activity (i.e. "Eat at")
Coordinates	The location of the activity
Image URL	The Image of the activity retrieved from Yelp API
Rating Image URL	The rating of the activity retrieved from Yelp API
Phone number	The phone number of the activity retrieved from Yelp API

4 EVALUATION

4.1 Micro benchmarks

Several tests and evaluations have been performed to the *Traveltant* system including the client application and the cloud server to ensure the reliability and the efficiency of the system.

4.1.1 Traveltant Client

Since the *Traveltant* client is only responsible for the user interface, many tests have been performed to ensure the performance and reliability of the application. The *Traveltant* client has been tested in the Windows Phone 7 Emulator and in a Samsung Focus device, which has a 1GHz processor, a GPS chip, and a 4-inch screen with 480×800 pixels. The application has been verified to work well in different versions of Windows Phone including 7.0, 7.1, and 7.5.

Furthermore, *Traveltant* client has been evaluated utilizing Microsoft's Windows Phone Performance Analysis tool [22]. With this tool, the application has run 50 times, with the user already signed in, and Miami, FL has been chosen as the desired destination for a 5-day trip. The average values of performance factors are gathered as the following:

Test	Average Value	Description
Memory usage	37.2 MB	The maximum amount of phone memory being used by the application measured in megabytes
Bandwidth usage	126 KB	The amount of bandwidth consumed in kilobytes

4.1.2 Cloud Service

The cloud service response time has been evaluated using an average of 50 different requests of a 5-day trip, and the results come as the following:

Action	Average Time
Building a user profile	9.1 sec
Generating a personalized trip plan	3.4 sec

Building the user profile takes 9.1 seconds in average as the system adopts the disposable user profile approach; however, users do not notice this delay since building the user profile is performed in the background while the user selects the trip details. Facebook API is held responsible for this delay since it takes around 8.3 seconds to get back with all the requests, while the rest of this time is shared between *Traveltant* process and Yelp API.

Traveltant destinations database Yelp has also been tested, and it works perfectly in North America and European destinations where the Yelp website is popular among users. Nevertheless, in areas where Yelp is not popular such as some Asian countries, the results of the system are not accurate, and therefore, more databases could be integrated with the system to extend its functionality in these areas.

4.2 Macro benchmarks

4.2.1 Evaluation Methodology

Evaluating personalized search results is always a challenge since users can only evaluate the results themselves [9]. Every person can judge if the result fits his or her needs for every particular request or not, depending on many personal factors such as personal taste. As a result, automated methods cannot be used to evaluate such results accurately, and in consequence, direct users' feedback is essential for evaluation.

4.2.2 Experiment

To obtain a ground truth, 15 volunteers were asked to evaluate the system. They were asked to find and “like” at least five favorite restaurants' pages in their Facebook account.

Alternatively, they can “check in” their favorite restaurants by using their Facebook account.

Afterward, they were given two identical phones with two different versions of the system (as in Figure 10): personalized and non-personalized. The non-personalized version uses only generic factors for recommendation such as the popularity of the activity and its rating. Then they were asked to choose a destination in the U.S.A. and use it in both phones. After that, they were asked to rank each suggested activity in a 10-point scale without informing them which version is which as in Figure 11. Users can rank an activity 10/10 only if they think that this activity perfectly fits their interests.

Upon gathering the rating results, the rating average of the suggested activity in the personalized version was around 7.73/10, and in the non-personalized version was around 6.35/10 rating average. This result shows that using social network for personalization gives better recommendations; using only generic factors is not enough. On the other hand, activities generated by the non-personalized version were slightly better in Yelp rating (out of 5), since this rating is based on the popularity of the activity for all people without considering every particular user's interests as shown in Figure 13.

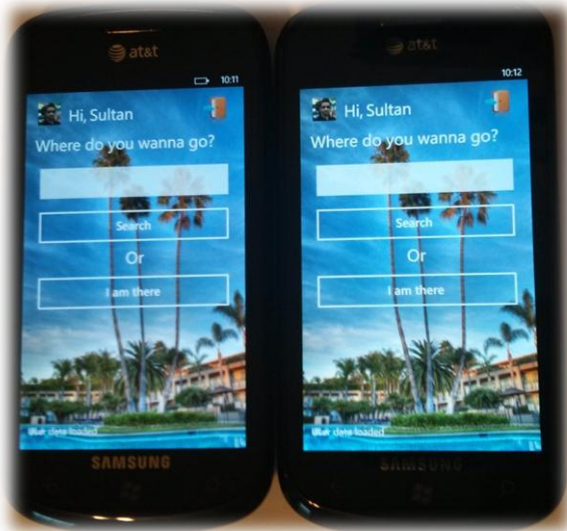


Figure 10: Phones used for results evaluation

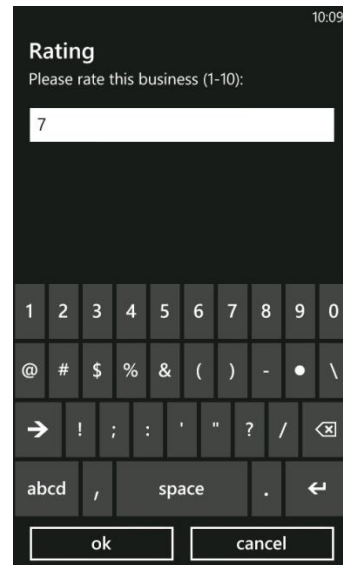


Figure 11: Activity Rating Window

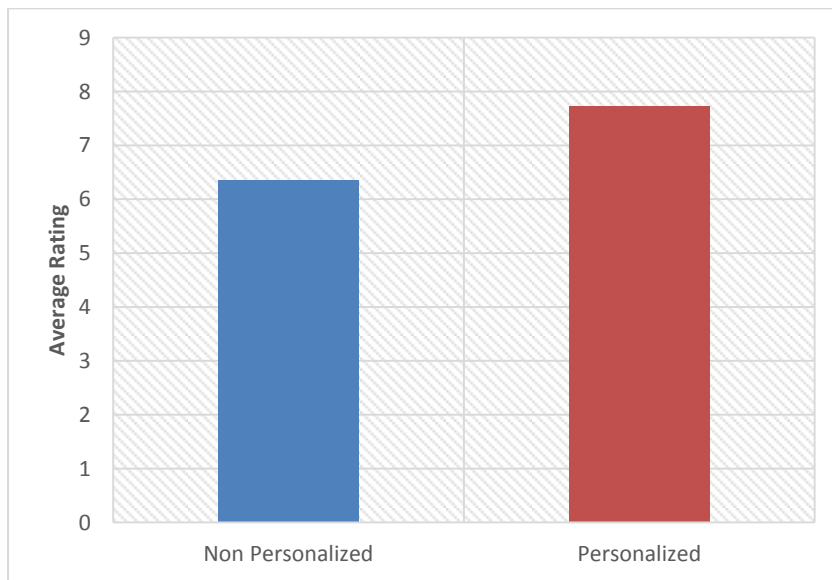


Figure 12: Activities rating average for both personalized and non-personalized versions.

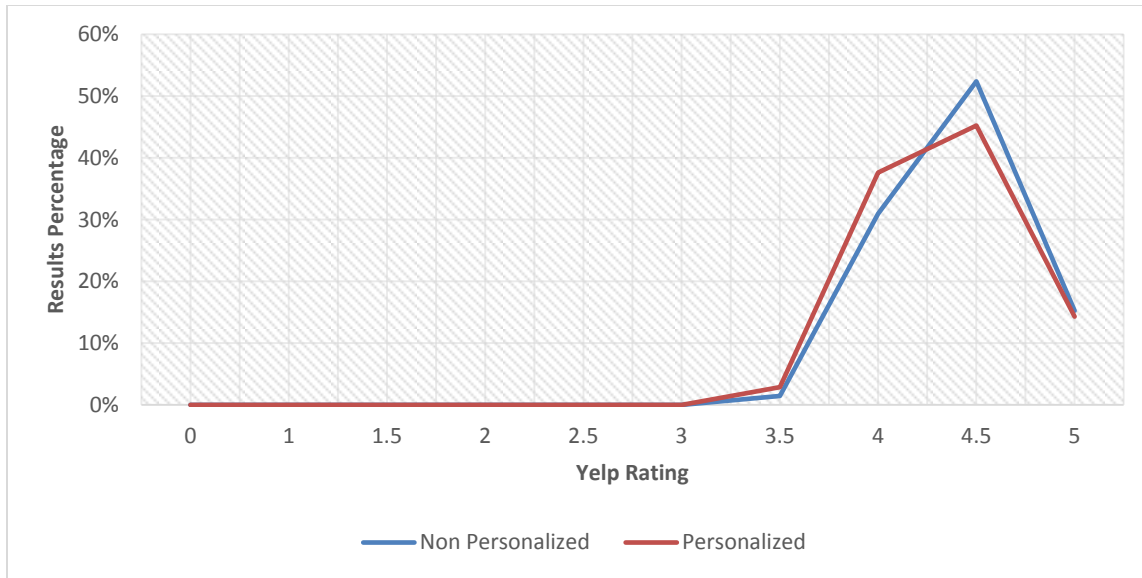


Figure 13: Activities Yelp rating

5 CONCLUSIONS

5.1 Summary

A social network based smart system that recommends personalized trip plans is proposed in this document. The system analyzes the user's social network and builds a user profile that contains the interests of this user in addition to his most interactive friends. The system assigns an interactivity score for every person in the user's friends list based on how much they have engaged in social network activities. Next, the system recommends a detailed personalized trip plan for the user using an implemented Windows 7 phone application while the rest of the work is implemented in a backend cloud service. In evaluating the results of the system, volunteers' rating shows that the personalized results are better than those results inferred from only common factors such as popularity and rating.

5.2 Future Work

There are several improvements anticipated for future work in this project. In the field of social networks based personalization, the friends interactivity list could be calculated using additional factors other than relying on the number of interactions only. One factor could be the distribution of friend interactivity temporally in order to distinguish between old and new friendships. Another enhancement could be performed to analyze the nature of the friend to determine if this friend's preferences are similar to the user's interests. For example, a sister can interact with her brother frequently in a social network, but her travel interests are different from her brother's. The system could identify such cases by further analysis of friends' natures and decide whether their interests resemble the user's or not.

Other travel related enhancements could be achieved by adding additional travel related criteria to the recommendation algorithm. One of these criteria is to consider the transportation

options of the user, and based on the available transportation method, the attractions list can be adopted to include those activities that can be reached using the selected transportation method. Similarly, considering the user's budget for every trip is also important, so if the user's budget is limited, activities available through affordable public transportation will be preferable. Moreover, an average historical weather data such as Weather Spark [23] could be a good indicator about the possible weather at the trip time, so the activity suggestion algorithm could consider this data to suggest the appropriate activity for that particular weather.

6 REFERENCES

- [1] Y. Zheng and X. Xie, "Learning travel recommendations from user-generated GPS traces," in *ACM Transactions on Intelligent Systems and Technology (TIST)*, New York, NY, 2011.
- [2] Facebook, "Facebook Reports Fourth Quarter and Full Year 2012 Results," Menlo Park, 2013.
- [3] V. Gundotra, "Google Official Blog," Google, 6 December 2012. [Online]. Available: <http://googleblog.blogspot.com/2012/12/google-communities-and-photos.html>. [Accessed 6 May 2013].
- [4] "Yelp for Developers," Yelp, [Online]. Available: www.yelp.com/developers/documentation. [Accessed 6 May 2013].
- [5] "Bing Developer Center," Microsoft, [Online]. Available: www.bing.com/developers. [Accessed 6 May 2013].
- [6] J. Pitkow, H. Schütze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar and T. Breuel, "Personalized Search," *Communications of the ACM*, vol. 45, no. 9, pp. 50-55, September 2002.
- [7] B. Tan, X. Shen and C. Zhai, "Mining Long-term Search History to Improve Search Accuracy," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, Philadelphia, PA, USA, 2006.
- [8] Z. Dou, R. Song and J.-R. Wen, "A Large-scale Evaluation and Analysis of Personalized Search Strategies," in *Proceedings of the 16th international conference on World Wide Web*, Banff, Alberta, Canada, 2007.
- [9] D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har'El, I. Ronen, E. Uziel, S. Yogev and S. Chernov, "Personalized Social Search based on the User's Social Network," in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, Hong Kong, China, 2009.
- [10] C. Wilson, B. Boe, A. Sala, K. Puttaswamy and B. Zhao, "User Interactions in Social Networks and Their Implications," in *Proceedings of the 4th ACM European Conference on Computer Systems*, Nuremberg, Germany, 2009.
- [11] E. Amitay, D. Carmel, N. Har'El, S. Ofek-Koifman, A. Soffer, S. Yogev and N. Golbandi,

- "Social search and discovery using a unified approach," in *Proceedings of the 20th ACM conference on Hypertext and hypermedia*, Torino, Italy, 2009.
- [12] J. Golbeck, "Generating Predictive Movie Recommendations from Trust in Social Networks," in *Proceedings of the 4th International Conference on Trust Management*, Pisa, Italy, 2006.
- [13] A. Seth and J. Zhang, "A social network based approach to personalized recommendation of participatory media content," in *Proceedings of The International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2008.
- [14] I. Guy, N. Zwerdling, I. Ronen, D. Carmel and E. Uziel, "Social media recommendation based on people and tags," in *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, Geneva, Switzerland, 2010.
- [15] I. Guy, N. Zwerdling, D. Carmel, I. Ronen, E. Uziel, S. Yogev and S. Ofek-Koifman, "Personalized recommendation of social software items based on social relations," in *Proceedings of the third ACM conference on Recommender systems*, New York, New York, USA, 2009.
- [16] H. Yin, C. Wang, N. Yu and L. Zhang, "Trip Mining and Recommendation from Geo-tagged Photos," in *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops (ICMEW '12)*, Washington, DC, 2012.
- [17] R. Ji, L.-Y. Duan, J. Chen, S. Yang, H. Yao, T. Huang and W. Gao, "Learning the trip suggestion from landmark photos on the web," in *Proceedings of the Image Processing (ICIP), 2011 18th IEEE International Conference*, 2011.
- [18] A.-J. Cheng, Y.-Y. Chen, Y.-T. Huang, W. Hsu and H.-Y. Mark Liao, "Personalized travel recommendation by mining people attributes from community-contributed photos," in *Proceedings of the 19th ACM international conference on Multimedia*, Scottsdale, Arizona, 2011.
- [19] B. Schmidt-Belz, A. Nick, S. Poslad and A. Zipf, "Personalized and Location-based Mobile Tourism Services," in *Proceedings of the Fourth International Symposium on Human Computer Interaction with Mobile Devices*, Pisa, Italy, 2002.
- [20] A. Echtibi, M. J. Zemerly and J. Berri, "Murshid: A Mobile Tourist Companion," in *Proceedings of the 1st International Workshop on Context-Aware Middleware and Services: Affiliated with the 4th International Conference on Communication System Software and Middleware (COMSWARE 2009)*, Dublin, 2009.

- [21] "Facebook Developers," Facebook, [Online]. Available: developers.facebook.com. [Accessed 6 May 2013].
- [22] "Windows Phone Performance Analysis," Microsoft, [Online]. Available: <http://msdn.microsoft.com/en-us/library/hh202934%28v=VS.92%29.aspx>. [Accessed 12 May 2012].
- [23] "WeatherSpark," Vector Magic, Inc, [Online]. Available: www.weatherspark.com. [Accessed 6 May 2013].
- [24] "Facebook Query Language," [Online]. Available: http://en.wikipedia.org/wiki/Facebook_Query_Language. [Accessed 29 April 2012].