PERSONALIZED TIME-AWARE OUTDOOR ACTIVITY RECOMMENDATION SYSTEM

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SHAKIBA RAHIMIAGHDAM

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

AUGUST 2015

Approval of the thesis:

PERSONALIZED TIME-AWARE OUTDOOR ACTIVITY RECOMMENDATION SYSTEM

submitted by **SHAKIBA RAHIMIAGHDAM** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver ___________________
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Adnan Yazıcı ___________________
Head of Department, **Computer Engineering**

Assoc. Prof. Dr. Pınar Karagöz ___________________
Supervisor, **Computer Engineering Dept., METU**

Assist. Prof. Dr. Alev Mutlu ___________________
Co-supervisor, **Computer Engineering Dept., Kocaeli Uni.**

Examining Committee Members:

Prof. Dr. Nihan Kesim Çiçekli ___________________
Computer Engineering Dept., METU

Assoc. Prof. Dr. Pınar Karagöz ___________________
Computer Engineering Dept., METU

Assist. Prof. Dr. Alev Mutlu ___________________
Computer Engineering Dept., Kocaeli Uni.

Prof. Dr. İsmail Hakkı Toroslu ___________________
Computer Engineering Dept., METU

Prof. Dr. Ahmet Coşar ___________________
Computer Engineering Dept., METU

Date: ___________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: SHAKIBA RAHIMIAGHDAM

Signature :

ABSTRACT

PERSONALIZED TIME-AWARE OUTDOOR ACTIVITY RECOMMENDATION SYSTEM

Rahimiaghdam, Shakiba

M.S., Department of Computer Engineering

Supervisor: Assoc. Prof. Dr. Pınar Karagöz

Co-Supervisor: Assist. Prof. Dr. Alev Mutlu

August 2015, 77 pages

The new growing generation of the communication technology has been gaining enormous popularity in the past few years. Location-based social network as one of the platforms in this field, has been providing services and facilities to enhance user experience to explore their surrenders and new places. Among current services, point of interest (POI) recommendation and activity recommendation draws significant attention of users, which makes it a potential field of the study. However, despite of all the developments performed in this field, activity recommendation system still requires further improvements, since only a few related studies concentrated on this topic so far.

In order to develop the activity recommendation system, we present two approaches using different ideas by extending existing location-based collaborative filtering (CF) recommendation models. One of them focuses on the temporal feature of the data and the other one emphasizes on the correlation between activities to estimate the probability of selecting each activity. We evaluated our systems on a medium-scale real data set gained by the combination of the Gowalla and Foursqaure. The experimental results confirm that both our proposed methods remarkably outperform the basic CF model. In addition, we study several extensions on the location-based techniques as the minor contributions of this thesis.

Keywords: Recommendation System, Collaborative Filtering Technique, User-based Collaborative Filtering Technique, Activity Recommendation, Time-Aware Activity Recommendation, Activity Correlations, Location-Based Recommendation

ÖZ

ZAMANA GÖRE KİŞİSELLEŞTİRİLMİŞ DIŞ AKTİVİTE ÖNERİ SİSTEMİ

Rahimiaghdam, Shakiba

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi: Doç. Dr. Pınar Karagöz

Ortak Tez Yöneticisi: Yard. Doç. Dr. Alev Mutlu

Ağostos 2015, 77 sayfa

Yeni nesil iletişim teknolojisi son birkaç yıl içerisinde büyük bir ilerleme ve popülerlik kazanmıştır. Bu alandaki platformlardan biri olan konum-bazlı sosyal ağ, kullanıcılara sunduğu yeni hizmet ve olanaklarla, onların deneyimlerini geliştirerek, çevrelerini ve yeni yerler keşfetmelerini sağlamaktadır. Bu hizmetler arasında, ilgi çekici yer önerisi ve aktivite önerisi kullanıcıların ilgisini çekmektedir, bu da potansiyel bir araştırma alanı ortaya çıkarmaktadır. Bu alanda görülen tüm ilerlemelere rağmen, bu konuda çok az bilimsel çalışma yapılmıştır ve aktivite öneri sisteminin hala geliştirilmesi gerekmektedir.

Aktivite öneri sistemini geliştirmek için, bu çalışmada mevcut konum bazlı işbirlikçi filtreleme sistemi geliştirilerek ve farklı görüşler kullanılarak iki farklı yaklaşım tanıtılmıştır. Bunlardan ilki verilerin zamansal özelliği (üzerinde dururken, diğeri ise her bir faaliyeti seçme olasılığını tahmin etmek için aktiviteler arasındaki korelasyonu vurgular. Bu çalışmada, Gowalla ve Foursqaure kombinasyonu ile elde edilen orta ölçekli gerçek veri seti üzerinde bu iki sistemin değerlendirilmesi yapılmıştır. Deneysel sonuçlara baktığımızda, önerilen iki sistemin de temel davranışsal filtreleme sistemine göre çok daha iyi olduğu görülmüştür. Bunlara ek katkı olarak, bu çalışmada konum bazlı tekniklerde geliştirilmiştir.

Anahtar Kelimeler: Öneri Sistemi, Davranışsal Filtreleme Tekniği, Kullanıcı-Bazlı İşbirlikçi Filtreleme Tekniği, Aktivite Önerisi, Zamana Göre Aktivite Önerisi, Aktivite Korelasyonu, Konum Bazlı Önerisi

To My Dear Parents, Fariba & Rahim

And

To My Lovely Grandmother Who Passed Away Peacefully

But Her Memories Always Stay With Me

ACKNOWLEDGEMENTS

This thesis would not have been possible without the contribution of many people. I am thankful for all of them that they never left me alone. First of all, I wish to express my deepest appreciation to my dear supervisor, Assoc. Prof. Dr. Pınar Karagöz. Your unconditional support, inspiring guidance, trust, encouragement, and expertise through this difficult process have made the completion of this master thesis possible. One simply could not wish for a better or friendlier supervisor and you will be always my role model in real and academic life.

To my caring, supportive, and loving mother Fariba Derakhshanfar and father Rahim Rahimiaghdam, I would like to thank you sincerely for your everlasting effort for my success. Your sacrifice throughout my life is inexpressible. I always owe a debt of gratitude to you because of believing and unconditional loving me.

To my dearest friends, Masoud Sattari, and Saeid Sattari, I would like to express my heartfelt appreciation to you for being present whenever I need. Your generous support and recommendations through this process was quite valuable for me. Big thanks to you for being truly my best friends.

To my beloved, Shahram Mollahasani, the completion of this thesis could not have been accomplished without your endless support, patience, and help. Any words cannot express my intense feelings for you. You believed in me even when I did not believe that I could achieve. Whenever I need, you have been with me with your never-ending love. The eternal appreciation goes to you for enriching and shedding light on my life beyond all expectations.

TABLE OF CONTENTS

CHAPTERS

APPENDICES

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

POI	Point of Interest
RS	Recommendation System
CF	Collaborative Filtering
LBSN	Location-Based Social Network
MTRS	Modified Time-Aware Recommendation System
MACRS	Modified Activity-Correlated Recommendation System
CB	Content-Based
CDR	Call Data Record
GPS	Global Positioning System
RFID	Radio Frequency Identification
BRS	Basic Recommendation System
BS	Basic Similarity
PIP	Proximity Impact Popularity
NHSM	New Heuristic Similarity Model
PSS	Proximity Significance Singularity
JPSS	Jaccard PSS
URP	User Rating Preference
TRS	Time-Aware Recommendation System

TS	Time-Aware Similarity
ATS	Advanced Time-Aware Similarity
ATRS	Advanced Time-Aware Recommendation System
SRS	Space-Aware Recommendation System
URS	Unified Recommendation System
UTRS	Unified Time-Aware Recommendation System
ACRS	Activity Correlation-Aware Recommendation System
TACRS	Time-Aware Activity-Correlated Recommendation System
UACRS	Unified Activity-Correlated Recommendation System
API	Application Program Interface
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative
DB	Day-Based
HB	Hour-Based

CHAPTER 1

INTRODUCTION

Dealing with data has always been a hot topic for computer experts since the size of the data which should be handled is growing enormously as time passes. At the beginning, the challenge was how to store and retrieve data, but at present, the matter is how to use the stored data in a beneficial way. In fact, it is proved that lots of information more than what we are able to see can be extracted from those data which is a great advantage. By gaining that information in the related field, we can experience impressive advancements.

Different approaches and systems have emerged in order to enhance the quality of data usage. The World Wide Web as the largest data repository, and the web search engines as the tools for assisting users to search and find necessary information, can be considered as one of the most famous examples for the information retrieval improvements. Google[1], which is one of the most popular and well-known search engines, aims to fulfil the users' requirements, utilizing various techniques to finding appropriate results to their queries through billion pages recorded in the data corpus of internet [1].

As technology of information science improved, a new generation of systems, called as recommendation systems have been raised. They can be considered as one of the smart versions of the information processing systems, since they are able to transcend the traditional concept of information retrieval and even facilitating it by offering personalized information automatically. The idea is that the requirements or

[1] www.google.com

[2] www.loopt.com

interests of the users can be recognized in advance and in the suitable time they can be recommended to users respectively.

These systems detect user preferences by collecting information directly from the user or extracting it implicitly, from the history of the user behavior. Then automatically suggest appropriate items or activities regarding the domain of the system to the user by applying different techniques. Recommending books, software, electronics and many other types of products and retails from the Amazon web site [2], or movies by MovieLens [3], are popular examples of different domains for recommendation systems.

From the mid-1990s, in which recommendation systems appeared as an important research area, many studies have been performed in order to develop new techniques in this field [4] [5] [6]. Although, remarkable progresses have been done so far, still there is a high demand for new innovations, due to the richness of this area and the vast application of recommendation systems in daily life of people.

The rapid growth of the mobile devices related technologies such as wireless networks or GPS, leads to the formation of the location-based social network (LBSN) phenomenon such as loopt[2], Brightkite[3], Foursquare[4], Gowalla[5], and Whrrl[6] which create consequently the opportunity to check-in at different places as a new trend for mobile users [7] [8]. It also allows users to leave comments, tips, or giving rates based on the satisfaction degree of the visited location or activity performed. All these facilities accelerate the large data accumulation growth day by day, which is the key to boosting the knowledge of the user characteristics, and correspondingly,

[2] www.loopt.com

[3] www. brightkite.com

[4] www. foursquare.com

[5] www.gowalla.com

[6] www.whrrl.com

assisting experts to design useful services for users to enhance the modern life of today.

Beside the point of interest (POI) recommendation which has been drawing significant attention, whether in academic studies or in industry [9] [10], various services are developed based on the LBSN data which seem to be attractive to users as well. Recommendation systems for outdoor activities such as playing bowling or going to an American restaurant, can be counted as one of these services we can point to. With the increasing the variety of the activities can be done in spare time such applications are definitely useful for people to have many new experiences and enjoy their limited free times of today's modern life.

1.1 Motivation and Contributions

To the best of our knowledge, there are unfortunately only a few studies have been accomplished regarding the activity recommendation methods. Henceforth we believe our work can be categorized as a novel work among current single-feature derived activity-based recommendations. The contributions of this work are declared as follows:

- We propose two main frameworks using the techniques presented in [11] for personalized activity-based recommendation exploiting extended temporal features and we evaluate their accuracy of the data gained by combining the Gowalla and Foursquare. The first one emphasizes on including the temporal properties of the data and the other one, in addition to temporal features, concentrates on the correlation between activities. For each model we present three approaches; Hour-based, day-based and the integrated day-hour-based as the unified system due to the importance of the time influence on the type of activity user intends to do.
- We also expand our study by examining the effect of two different similarity algorithms on the performance of the basic collaborative filtering recommendation.

- Besides these, two hybrid location recommendation systems for combining the two existing time-based and space-aware techniques from the work [11] is offered and evaluated on the location-based Gowalla data set.

1.2 Thesis Outline

This thesis report is organized as follows: In Chapter 2, we study the background and related work about recommendation systems. We aim to review the highlighted recommendation approaches in the literature regarding the techniques have been modeled on the LBSN data by extracting different features such as social ties or temporal factors.

The preliminaries to our work presented in Chapter 3. Since we have been inspired by the location-based recommendation framework in the [11], related methods from the mentioned work are explained comprehensively. In addition, two minor extensions which have been performed regarding hybrid location-based recommendation are presented in this chapter. Meanwhile, two algorithms for measuring user similarities applicable on user-based CF approach are introduced as well.

In Chapter 4, our main contribution is presented towards two main approaches. The Modified Time-Aware Recommendation System (MTRS), and the Modified Activity-Correlated Recommendation System (MACRS) as activity-based recommendation systems applicable on day, hour, and the combination of both as temporal factors.

Chapter 5 is dedicated to the experiments and the evaluation of the proposed methods. The data set characteristics and the evaluation metrics applied in this thesis are explained. Meanwhile, the results of all mentioned methods are shown by graphical charts in this chapter.

Lastly, the thesis work is concluded with an overview and future work in Chapter 6.

CHAPTER 2

RELATED WORK

Paul Resnick et al. in the early work [12] wrote "It is often necessary to make choices without sufficient personal experience of the alternatives. In everyday life, we rely on recommendations from other people either by word of mouth, recommendation letters, movie and book reviews printed in newspapers, or general surveys such as Zagat’s restaurant guides".

A recommendation system (RS) as mentioned in the previous chapter, can be considered as an information process, which builds recommendation on top of users and items data. According to the comprehensive work of [13] different RS techniques would be compared with each other based on the methodology and knowledge source factors. On one side by extracting interests and characteristics of users and on the other side, by gathering the features of items, the main goal is to recommend fairly appropriate new items to the user. Based on the type of the data and the source of the knowledge which is used by the RS the techniques would be varied [14].

In this chapter first of all, the major well-known recommendation system techniques which can be divided into three groups of approaches: collaborative filtering (CF), content-based filtering (CB) and hybrid methods are described briefly. Then the Location Based Social Network concept is defined based on the studies in the literature, and in the following, the researches have been performed regarding the two popular recommendation services such as location recommendation and activity recommendation is reviewed.

2.1 Collaborative Filtering

One of the promising techniques which have been used mostly in recommendation context is collaborative filtering [15] [16]. The philosophy behind this approach leverages the famous "wisdom of the crowd" belief. For better explanation, CF recommends users the high ranked items of the most similar users to recommend them to the selected user. In other words, the more a user has a similar history to the selected user for what she did or what she likes, the more it is possible these two users have common interests in the future. So by this logic, the aim is filtering irrelevant items from a large collection by dint of collaboration between users, and that's why this technique is called collaborative filtering [17].

For describing the RS techniques it can be assumed there is a user-item matrix with each cell delimits the level of a user's interest to a specific item by giving ratings to them. The more the user-item collection size increases the more it is improbable to have rates for all the matrix cells. In this regard if we look at another aspect the substantial task of a recommendation system can then be defined as; estimating unrated items for a specific user and consequently recommend those with higher ratings to the user [18].

Meanwhile cold start and sparsity are the two weaknesses, have been challenging the efficiency of CF systems [19]. Cold start refers to the problem of lack of information for the new users joined to the system, which leads to the system failure for recommending suitable items to the new user. Sparsity also happens usually in typical RS, due to the large number of users and items registered on the system. Since each user only gives rates for a few items, detecting user interests completely in one side, and finding real similar users in another side have become a hard task.

The existing CF approach has been classified in three methods of user-based, item-based and model-based, in terms of the similarity model as discussed in [20]. User-based technique presented by Grouplens research for the first time in [5]. In this algorithm nearest users are recognized by applying user similarity methods. But in item-based which introduced in [21] items are the core of the similarity methods for the nearest users’ recognition. In the model based technique, based on the analyzing parameters of statistical models for user ratings, recommendations are obtained. For instance, we refer to one of the primary approaches discussed in [12] for mapping CF to a classification and creating a classifier for each active user for showing items as features over users and available ratings as labels and conjunct it with dimensionality reduction techniques to prevent data sparsity.

2.2 Content-Based Recommendation

Content-based algorithms for recognition of the preferred items of a particular user, concentrate on the item's properties solely as defined elaborately in [22]. By another interpretation, content-based systems perform analysis on both user profiles and item descriptions, as a means to find the most matched items for a specific user by comparing both items and users' interests [23]. In a sense, the cold start problem has been solved since because of having user profiles and item profiles there is no need to have a history of the user's feedback history necessarily, which is a strength point of this approach in comparing to the CF. But in another sense, the items nominated for suggestion always have the same properties. So the chance of trying new varying items is not given to the user.

2.3 Hybrid Recommender Systems

Hybrid systems as stated by Bruke in [24], consists of two or more techniques which have been fused to cover their rough edges and reinforce their strengths. It can be remarked that it is also possible to mix two CB or two CF recommender systems which belong to the same category as a hybrid method [25]. In previous sub-sections we explained only two general categories in this area, however, there are many branches of RS techniques which has been presented so far which are not mentioned in this work. There are divergent ideas for how to combine different techniques for building a hybrid system. Some proposed methods have been summarized in Table 1 [15] as it is shown in the below:

Table 1. Mixturing method for hybrid recommendation system

Mixture method	Description
Weighted	The scores (or votes) of several recommendation techniques are combined together to produce a single recommendation.
Switching	The system switches between recommendation components depending on the current situation.
Mixed	Recommendations from several different recommenders are presented at the same time.
Feature Combination	Features extracted from different data sources are combined together and given to a single recommendation algorithm.
Feature Augmentation	One recommendation technique is used to compute a feature or set of features, which is then part of the input to the next technique.
Cascade	Recommenders are given strict priority, with the lower priority ones breaking ties in the scoring of the higher ones.
Meta-level	One recommendation technique is applied and produces some sort of model, which is then the input used by the next technique.

2.4 Point Of Interest Recommendation

POI recommendation as one of the popular fields of research in recommendation system areas assists users to experience new places and explore surrenders better depends on their preferences and interests [26]. Users willingness can be inferred by direct opinions which gained by the explicit rating like five scale ratings, or deduced from the behavior of the user observed in a period such as frequent check-ins at a specific restaurant which can be interpreted as the user’s positive opinion about that particular restaurant [17]. The location entity can be defined by absolute or relative or symbolic formats as longitude and latitude coordinates, geographical cardinal directions and degrees, or the home or office icon respectively. In this way it is provided that users, photos, comments, locations and other elements of the network linked together [27] [28].

J. Bao et al. in [29] categorize the present location recommendations in two types of generic POI recommendation and personalized POI recommendation. The first type refers to techniques which process all of the data regardless of an individual user preference, in order to extract the most popular venues or places in a city for recommending to all users. For instance, in [30] authors apply the HITS algorithm to predict the popularity scale of a venue which has been built on the patterns drawn by mining a large data of the location and travel sequences. Personalized POI recommendation as the second type, emphasizes on producing recommendations per users separately under the characteristics and behavior of each user. It is attracting remarkable attention, due to the usability it has in mobile and web based services and advertisements such as coupon delivery [31].

Unfortunately the traditional pure collaborative filtering techniques are not able to perform efficiently, specifically for the LBSN data for two reasons: Firstly, LBSN data set is sparse, due to lack of enough record for each user. Particularly, in comparison with GPS trajectory data recorded every 5 seconds for instance, or Call Data Record (CDR) [32] logged every time user switches from the supporting cell phone antenna area to another one, a few check-ins are usually available. Secondly, there is no access to users to ask for more information and discover the user real opinion. Even at first glance, it may appear that there are insufficient numbers of fields in this dataset. In general, for basic data sets, time stamp information of the check-ins and physical location coordinates related to each user check-in are given as the fields. For these reasons we are interested to introduce the work discovered and analysed more features rather than basic user and location raw data to improve the performance of the CF techniques.

Cho et al. [33] by analyzing two types of data sets, LBSN data and a cell phone trace of 2 million people, find out that 10% to 30% is the share of social ties and 50% to 70% is the share of periodic behavior in all human mobility pattern. Henceforth, they develop a parameter estimation-based model by integrating both periodic and social mobility factors to predict the user location at any time with the accuracy of 40%. Wang et al. [18] propose two algorithms considering not only the history of user check-ins and the similarity between users, but also the friendship among them and the physical location of the venues. The initial idea comes out from the graph coloring algorithm.

A personalized recommendation framework is introduced in [29] which consider both geographical situation of the user and the expert suggestions. The model can be divided into two as offline and online parts. The user's preferences and behavior are extracted in the offline model as well as the user's expertise percentage for each city regarding to the location categories frequency in their histories. In the online section, with respect to the current geographical situation of the user and both the expert opinions and user's preferences information acquired by mining the procedure in the offline part, the new venues are picked and recommended to a user. The cold start problem for new users or for users visit a new city for the first time is tackled by adding the expert idea, and mapping locations of activities leads to reduce the sparsity of the data and find similar users without considering whether they are from the same city or not.

Regarding to efforts that have been done through the work in [34] the high dependency of the check-in dynamic and the day of the week as a temporal feature and the distance as a geo-related feature is studied. Based on this empirical research which is conducted on a large scale sample of Foursquare data set the activity trend among weekdays and weekends and the hours of the day is analyzed in addition to other temporal patterns such as the activity transition time interval and the how the check-ins possibly occurs in terms of time or physical distance.

Although it is proven that there is a strong relation between the locations and the time that check-ins happen in them, unfortunately there is a limited number of studies that have focused on temporal feature of the ratings for computing recommendation scores in CF. As the emphasize is on the influence of temporal incorporation in [35], the author denotes that using the old data and new data with an equal weight has no different than choosing items blindly. Actually, the trials have been carried out in this area can be separated into two classes; one refers to studying the temporal characteristic of the items, and the other one refers to investigating the temporal characteristic of the users and the time the check-ins happened.

As one of the early studies in [16] a novel clustering-based algorithm has been proposed for weighting items based on the time they were selected. In fact the items have been purchased or chosen recently, logically should have more priority rather than the items in the older history of users. The reason is the users' interests can change, hence considering this feature improves the item-based performance. However, Yuan et al. in [11] present a time-based method which is totally different from the mentioned works in terms of the time-factor. In [11] which is the work we have extended, the periodic temporal characteristic is considered by splitting a large time interval such as day into periodic slots such as hour. In contrast, the previous studies used the gap between the time that a check-in occurred at, and the given time for a recommendation as a decaying parameter for weighting. It is noteworthy to know that the sparsity of the user-time-location cube is reduced through applying a smoothing method in the proposed work of [11].

2.5 Activity Recommendation

Activity recommendation is almost a new research area. There exists variation among the activity recommendation studies due to the different ways that the term activity can be defined. As a start in this field, most of the work focused on prediction of the users activity in a specific time by the sensor data or GPS data [36]

[37]. The work in [38], by processing RFID data, aims to recognize daily indoor activities such as watching TV or taking a shower. The work in [8] tries to step further and recommend activities if users intend to go to specific places and reversely recommend locations if they prefer to perform specific activities. Based on our research, except location-activity correlation works and activity transition probability estimation studies, as the most popular research topics related to activity recommendations, there is no CF recommendation for outdoor entertainment activities based on a single user interest with respect to a given time parameter exists or developed in the literature.

CHAPTER 3

LOCATION-BASED RECOMMENDATION SYSTEM CONCEPTS AND EXTENTIONS

This chapter presents the basic concepts related to location-based recommendation systems as an introduction to our main contribution which is introduced in the next chapter. Meanwhile, some minor extensions which have been carried out, are expounded with the aim of being a motivation for the future work. First of all, the very basic collaborative filtering recommendation technique will be presented. There are two extensions of the basic method which have been performed by different similarity algorithms. In the next section, the methods given in [11] will be explained as the preliminaries to our proposed framework; which are time influence incorporated approach, spatial influence incorporated approach, and lastly the unified method. As the final section, the two extensions of the unified framework will be offered.

3.1 Basic User-Based Collaborative Filtering Recommendation

Simple user-based or user-user collaborative filtering is the primal method among the automatic recommender system methods since GroupLens Usenet article described it for the first time [5]. The description is so perceptible and straightforward: In order to find a new item to recommend to the target user, firstly a collection of users which have a similar history with respect to the items rated to the target user should be identified. This collection can be called similar users. Then, based on their taste and their ratings, the highest ranked items from the most similar users will be selected as the top suggestions to the target user [39].

In a location-based recommendation system, if a group of people had similar taste in a period of time for what locations or places they went, it is highly probable that they would have common interests in the future too, which makes it easier to even guide each other for better decisions.

For this algorithm, it can be assumed the data is structured in a 2-dimensional matrix, rows indicating user IDs and columns indicating location IDs. The values are all zeros or ones; If the confluence of row x with column y is zero, it means that location y hasn't been seen by user x yet, or there is no available rating for location y by user x. Otherwise, if it is one, we can understand user x checked-in at location y once, at least. So the whole matrix contains only two values of ones or zeros.

For a given user u, and a specific location *l*, the Basic Recommendation System formula ($BRS_{u,l}$), given in Equation 3.1 works by weighting the check-in frequencies $c_{v,l}$ of the similar user set v, for the specific location *l*. Similar users will be distinctive by higher $BS_{u,v}$ values. The *Basic Similarity* function ($BS_{u,v}$) given in Equation 3.2, calculates the cosine similarity between the two vectors of common locations for every pair of users u, v, by considering the check-in values for the location *l*, $c_{u,l}, c_{v,l}$ in order to find most similar users and weight them by the percentage of the similarity. Among different metrics [40] [41] [39] cosine similarity is one of the most popular, well-to-do and useful methods in terms of accuracy for finding similarities in CF approach.

$$BRS_{u,l} = \frac{\sum_v BS_{u,v} c_{v,l}}{\sum_v BS_{u,v}} \tag{3.1}$$

$$BS_{u,v} = \frac{\sum_l c_{u,l}\, c_{v,l}}{\sqrt{\sum_l c_{u,l}^2}\, \sqrt{\sum_l c_{v,l}^2}} \tag{3.2}$$

The set of locations, which the recommendation score is computed for, is all the locations for each user that are not available in the history of that user. In the prediction we can estimate the chance of all locations (visited or not visited) for the next selection by the same user, but in the recommendation the main focus is on suggesting new items to the user. In fact, it is the most significant difference between prediction and recommendation. Since our goal is developing a recommendation system, we want to propose new, untried POIs to the user. So if we assume L as the set of all locations, and L_u as the set of locations belongs to the history of user u, L-L_u is the collection of locations which recommendation scores will be produced for by the Basic Recommendation System.

3.1.1 Proximity Impact Popularity (PIP) Similarity-Based Extension

For finding similar users in the CF recommendation system, varied algorithms exist. In the previous section, as mentioned, for calculating the *Basic Similarity* (*BS*), cosine similarity as one of the common and efficient methods is applied. In this subsection and the next one we will introduce two other similarity measures, Proximity Impact Popularity (PIP) and a New Heuristic Similarity Model (NHSM), respectively. The PIP similarity measure which is introduced in the following, has been composed by the multiplication of three parameters; *Proximity*, *Impact* and *Popularity*. It can be interesting to note that the name of PIP also has been created by abbreviating the name of three mentioned parameters.

In order to be able to explain the three mentioned factors, first of all we should define another parameter called the *Agreement*. It is a Boolean type variable, which takes true or false as the value. The Agreement between any pair of ratings r_1 and r_2 for an item, *Agreement* (r_1, r_2), will be considered true, if both ratings are on the same side of the median value of the rating scale. It means that ratings r_1 and r_2, should be both ″greater than or equal″ or ″less than or equal″ to median to be in agreement with each other. To emphasize, median, R_{med}, is calculated by first summing the minimum rate, R_{min}, and maximum rate, R_{max}, in the rating scale up, and then divide them by two. Now we are able to define the three main factors as given in Equations 3.3, 3.4, 3.5 and 3.6.

$$D(r_{1,}r_2) = |r_1 - r_2| \qquad ;if\ Agreement(r_{1,}r_2) = true$$

$$D(r_{1,}r_2) = \ 2\,.|r_1 - r_2| \quad ;if\ Agreement(r_{1,}r_2) = false \tag{3.3}$$

$$Proximity(r_{1,}r_2) = \left\{\{2 \,.\, (R_{max} - R_{min}) + 1\} - D(r_{1,}r_2)\right\}^2 \tag{3.4}$$

For calculating *Proximity*, firstly, the distance between two ratings $D(r_{1,}r_2)$, should be calculated. If the two ratings are in agreement, or in other words, the ratings vote for liking the item or both vote for disliking it, their distance will be the simple mathematical difference between two numbers. Conversely, if the ratings are in disagreement, which means one of the ratings is positive and the other one is negative, the distance is taken penalty by multiplying by two. Then the *Proximity* calculates the absolute difference for the pair of ratings.

$$Impact(r_{1,}r_2) = (|r_1 - R_{med}| + 1)(|r_2 - R_{med}| + 1)$$

$$;if\ Agreement(r_{1,}r_2) = true$$

$$Impact(r_{1,}r_2) = \frac{1}{(|r_1 - R_{med}| + 1)(|r_2 - R_{med}| + 1)}$$

$$;if\ Agreement(r_{1,}r_2) = false \tag{3.5}$$

The *Impact* factor shows the amount of users' willingness or unwillingness for an item. As an example, in a case that there are two pairs of ratings (3, 3) for r_1 and r_2 and (5, 5) for r'_1 and r'_2, despite the distance is zero and hence the *Proximity* is the same, the *Impact* value for the r'_1 and r'_2 ratings will be higher since they show a stronger preference on a common item.

$$Popularity(r_{1,}r_2) = 1 + \left(\frac{r_1 + r_2}{2} - \mu_k\right)^2$$

$$;if\ (r_1 > \mu_k\ and\ r_2 > \mu_k)\ or\ (r_1 < \mu_k\quad and\quad r_2 < \mu_k)$$

$$Popularity(r_{1,}r_2) = 1 \qquad ;otherwise \tag{3.6}$$

Where μ_k is the average of the all users' ratings for the item k, the *Popularity* factor has been modelled to augment the similarity value of the pair of ratings in the agreement which are further from the average rating. The average rating as the name reflects, shows a general fact about common ratings for an item, instead the more distanced ratings provide more information about the amount of two users' concurrence for a given item.

Finally, after computing mentioned factors, the PIP similarity of the two user *u* and *v* for a given item *i*, is calculated as given in Equation 3.7.

$$\begin{aligned} &PIP(r_{u,i}, r_{v,i}) \\ &= Proximity(r_{u,i}, r_{v,i}) \;.\; Impact(r_{u,i}, r_{v,i}) \;.\; Popularity(r_{u,i}, r_{v,i}) \end{aligned} \tag{3.7}$$

The PIP similarity of the two user *u* and *v* for all the items $i \in I$ is calculated by the summation of all the ratings for the two users on the all items belong to *I*.

$$sim(u,v)_{PIP} = \sum_{i \in I} PIP(r_{u,i}, r_{v,i}) \tag{3.8}$$

3.1.2 New Heuristic Similarity Model (NHSM) Similarity-Based Extension

The New Heuristic Similarity Model (NHSM), is an improved version of the PIP similarity method which aims to consider a wide range of factors rather than PIP. To put it differently, it is composed of other varied similarity measures which are introduced one by one as follow:

The first factor is the PSS similarity ($sim(u,v)_{PSS}$), which is calculated by the summation of $PSS(r_{u,i}, r_{v,i})$, parameter for every pair of user *u* and *v* of a common item *i* as given in Equation 3.9.

$$sim(u,v)_{PSS} = \sum_{i \in I} PSS(r_{u,i}, r_{v,i}) \tag{3.9}$$

In a similar manner, $PSS(r_{u,i}, r_{v,i})$, is also composed of the multiplication of the factors $Proximity(r_{u,i}, r_{v,i})$, $Significance(r_{u,i}, r_{v,i})$, and $Singularity(r_{u,i}, r_{v,i})$, for the given pair of users *u* and *v* and the common item *i*. It is defined as in Equation 3.10.

$$\begin{aligned} &PSS(r_{u,i}, r_{v,i}) \\ &= Proximity(r_{u,i}, r_{v,i})\,.Significance(r_{u,i}, r_{v,i})\,.Singularity(r_{u,i}, r_{v,i}) \end{aligned} \tag{3.10}$$

Similar to the *Proximity* in PIP method, here $Proximity(r_{u,i}, r_{v,i})$, also considers the distance between the two ratings. However the absolute distance matters here, regardless of the agreement concept, as given in Equation 3.11.

$$Proximity(r_{u,i}, r_{v,i}) = 1 - \frac{1}{1 + exp(-|r_{u,i} - r_{v,i}|)} \tag{3.11}$$

$Significance(r_{u,i}, r_{v,i})$, also restates that as the difference between the ratings and the median becomes greater, the ratings have more significance impression. In a case that $r_{u,i}$, is much higher than the median and, conversely $r_{v,i}$, is lower than the median impressively, it shows a clear contradiction among user *u* and *v* for item *i*, which leads to a lower similarity value. As for the case that both of the ratings are distanced from the median and close to each other, it can be concluded that there is a concurrence between the users indeed. Formula for *Significance* is given in Equation 3.12.

$$Significance(r_{u,i}, r_{v,i}) = \frac{1}{1 + exp(-|r_{u,i} - r_{med}|\,.\,|r_{u,i} - r_{med}|)} \tag{3.12}$$

The last factor is $Singularity(r_{u,i}, r_{v,i})$. It reflects the difference in the ratings $r_{u,i}$ and $r_{v,i}$, according to the average rating of item *i*, which is denoted as μ_i. Formula for *Singularity* is given in Equation 3.13.

$$Singularity(r_{u,i}, r_{v,i}) = 1 - \frac{1}{1 + exp\left(-\left|\frac{r_{u,i} + r_{v,i}}{2} - \mu_i\right|\right)} \tag{3.13}$$

All the three factors are as it can be seen are normalized, and it makes combining these factors with others possible, as they will be in the same range. To clarify, in the normalization process, the smallest value will be subtracted from the current value and the result will be divided by the range of the minimum value subtracted from the maximum one.

The other similarity measure which is deployed in the NHSM similarity is *Jaccard* factor, which is defined in Equation 3.14.

$$sim(u,v)_{Jaccard} = \frac{|I_u \cap I_v|}{|I_u| \times |I_v|} \tag{3.14}$$

For emphasizing proportion of the common ratings of the two users for the same item this similarity has been used. *PSS* similarity is combined with Jaccard by multiplying each other which is called *JPSS* that the formalization is shown in Equation 3.15.

$$sim(u,v)_{JPSS} = sim(u,v)_{PSS} \, . sim(u,v)_{Jaccard} \tag{3.15}$$

The last factor which is regarded is the general user behavior for given ratings to items, since some users tend to give higher rates and some others in contrast lower rates. The *URP* similarity in the formalization, which have been shown in Equation 3.16, applies the user preferences to the similarity measure by adopting the mean ratings μ_u, and μ_v, and the standard variances σ_u, and σ_v, of user u and v respectively as calculated by Equations 3.17 and 3.18.

$$sim(u,v)_{URP} = 1 - \frac{1}{1 + exp(-|\mu_u - \mu_v| \, . \, |\sigma_u - \sigma_v|)} \tag{3.16}$$

$$\mu_u = \sum_{i \in I_u} \frac{r_{u,i}}{|I_u|} \tag{3.17}$$

$$\sigma_u = \sqrt{\sum_{i \in I_u} \frac{\left(r_{u,i} - \bar{r_u}\right)^2}{|I_u|}} \tag{3.18}$$

Finally the NHSM similarity will be defined as multiplication of *JPSS* and *URP* similarities, as shown as follows:

$$sim(u,v)_{NHSM} = sim(u,v)_{JPSS} \,.\, sim(u,v)_{URP} \tag{3.19}$$

3.2 Time-Aware Model

The Basic Recommendation model which is introduced in the previous section, as the name expresses, is a very basic and primary model for the recommendation. Since it performs a very general comparison among users without regarding any specific feature such as time or other possible features rather than user ratings, definitely it has some drawbacks. Assume the case that the three users *i, j* and *k* have the same habit of going to place *x* for most of the time, but in different time hours. If user *i* and user *j*, have significant check-in counts for place *x*, around hour 4:00 pm, unlike user *k*, which is near 8:00 pm, then in the recommendation procedure user *j* and user *k* have the same priority for being a candidate to be recommended to user *i*. However, the users *i* and *k*, appears to be more similar in this case.

In order to address the indicated issue, one simple way is just injecting the time parameter to the Equations 3.1 and 3.2 to have Time-Aware Recommendation System ($TRS_{u,t,l}$) and Time-Aware Similarity ($TS_{u,v}$), respectively, as shown in Equations 3.20 and 3.21.

$$TRS_{u,t,l} = \frac{\sum_v TS_{u,v} c_{v,t,l}}{\sum_v TS_{u,v}} \tag{3.20}$$

$$TS_{u,v} = \frac{\sum_{t=1}^{T} \sum_{i=1}^{I} c_{u,t,l} \cdot c_{v,t,l}}{\sqrt{\sum_{t=1}^{T} \sum_{l=1}^{L} c_{u,t,l}^2} \sqrt{\sum_{t=1}^{T} \sum_{l=1}^{L} c_{v,t,l}^2}} \tag{3.21}$$

Due to time feature which is considered in the latter approach one more dimension is added to the data structure, hence two important points should be clarified:

First issue refers to sparsity. The user-location matrix which is gained from location based social network data normally is a sparse matrix, therefore after contributing time feature, the three-dimensional user-time-location matrix has much more unfilled cells. Obviously, check-in data for all users, for most of the locations at most of the time slots are not available. So by just incorporating time parameter in the basic similarity formula, because of data inadequacy, we may not to be able to indicate the users' similarity for a specific location visited in different time slots.

The other important point which causes the TS formula to fail from time incorporation is that in each loop of calculation for each two users' similarity, only one single time slot will be considered. As an example, if two people visited the same place, in different hours, but very close to each other like 6:00 p.m. and 7:00 pm, the similarity of these two will be zero. That is an inevitable imperfection of the mentioned formula which consider just the similarity calculation for the same time slots.

In order to overcome the mentioned issues the parameter ρ can be considered, which is calculated by Equation 3.22.

$$\rho_{t,t'} = \frac{\dfrac{\sum_{u=1}^{U} \sum_{l=1}^{L} c_{u,t,l} . c_{u,t',l}}{\sqrt{\sum_{u=1}^{U} \sum_{l=1}^{L} {c_{u,t,l}}^2} \sqrt{\sum_{u=1}^{U} \sum_{l=1}^{L} {c_{u,t',l}}^2}}}{U} \tag{3.22}$$

The ρ parameter reflects the time slots similarities based on the user check-ins. It is calculated by the cosine similarity between every pair of check-in vectors for each user $u \in U$ and location $l \in L$ at time t and t' for the total check-ins and averaged by the total number of user U. Accordingly, the adjacent time slots like 13:00 and 14:00 or 13.00 and 12:00 shows higher similarity with respect to check-ins behavior rather than distanced ones like 13:00 and 22:00.

By exploiting the ρ parameter, as shown in Equation 3.23, the main data cube is smoothed. The effect of time similarities will be exerted directly on the data to make the check-in behavior around close times more similar to each other. There is also one hidden benefit for this smoothing. We can estimate values for some zero cells by this extension. So the sparsity of the new cube will be decreased consequently.

$$\tilde{c}_{u,t,l} = \sum_{t'=1}^{T} \frac{\rho_{t,t'}}{\sum_{t''=1}^{T} \rho_{t,t''}} \, c_{u,t,l} \tag{3.23}$$

The Advanced Time-Aware Similarity formula (given in Equation 3.24) which uses the smoothed data can find similarities more accurate in comparison to the latter similarity functions which cause it to address the mentioned issues. By this equation, the problem of detecting similar patterns in unlike time slots check-ins almost has been solved. There are two scenarios which can happen: one is that a pair of user visited the same location at the same time, which brings the result about higher values of similarity, otherwise if two check-ins do not belong to the same time, bigger gap leads to lower similarity value and the smaller gap is converse.

$$ATS_{u,v} = \frac{\sum_{t=1}^{T} \sum_{i=1}^{I} \tilde{c}_{u,t,l} \cdot \tilde{c}_{v,t,l}}{\sqrt{\sum_{t=1}^{T} \sum_{l=1}^{L} \tilde{c}_{u,t,l}^{2}} \sqrt{\sum_{t=1}^{T} \sum_{l=1}^{L} \tilde{c}_{v,t,l}^{2}}} \tag{3.24}$$

Ultimately, the failed Time-Aware Recommendation System model has been improved to the Advanced Time-Aware Recommendation System as shown in

Equation 3.25, which calculates the recommendation score for a given user u, in a specific time *t*, for a candidate location *l* is:

$$ATRS_{u,t,l} = \frac{\sum_{v}(ATS_{u,v} \sum_{t'} \tilde{c}_{v,t',l} \cdot \rho_{t,t'})}{\sum_{v} ATS_{u,v}} \tag{3.25}$$

As it can be seen in the recommendation score formula (given in Equation 3.25) the ρ parameter is applied once again. It has a perspicacious application of this formula. In contrast to traditional recommendation approach, this time not only the items with the same check-in time is considered, but also the all untried items will be accounted either, and respectively weighted by the similarity cost between the time slot of t and t'.

3.3 Space-Aware Model

In the space-aware technique, the geographic position of the locations outweighs. Must be remembered that our data set provides two types of information; the chronological order of the check-ins associated with exact temporal details appears in the timestamp format, and the geographic information that indicates the precise locality of each check-in by the means of longitude and latitude coordinates. The previous section of this chapter focuses on the time-based feature, to enhance time-aware location-based recommendation systems. In this section we aim to introduce a method using the geographical information, to develop space-aware location-based recommendation systems.

The articles [11] [9] [42] are some of the samples which studied the POI distance impact on the user's mobility pattern. For visualizing the relation between distance and the check-in counts firstly, the distance between each pair of adjacent check-ins for all of the days is calculated. Only two contiguous locations within a day is considered as adjacent. As the next step, each distance is rounded by the second digit after the point and then the repetition of each distance category is counted. Finally, by aggregating results for all users the probability distribution function is plotted which illustrates the fact that the more the distance increases the more the check-ins probability decreases.

After discovering the reverse relation between distance and user interest for selecting a target POI, the degree of interest can be formalized by a function of distance in km as in Equation 3.26.

$$Interest\ \left(Distance(l_i, l_j)\right) = \ \alpha\ .\ Distance(l_i, l_j)^{\kappa} \tag{3.26}$$

The parameters of the Equation 3.26 can be easily approximated by maximum likelihood estimation. The probability that a user selects a location l_j while the initial location is l_i can then be computed by Equation 3.27.

$$p(l_j/l_i) = \frac{Interest\ \left(Distance(l_i, l_j)\right)}{\sum_{l_t \in l_u, l_t \neq l_i} Interest\ \left(Distance(l_i, l_j)\right)} \tag{3.27}$$

Based on the history of the user u, and the location history set l_u, utilizing the naïve Bayesian method the Space-Aware RS model for a new location l, can be computed as given in Equations 3.28 and 3.29.

$$SRS_{u,t,l} = p(l|I_u) = \ p(l) \prod_{l' \in I_u} p(l'|l) \tag{3.28}$$

$$p(l) = \beta\ \frac{|CF_{l,t}|}{\sum_{l' \in L} |CF_{l',t}|} + (1 - \beta)\ \frac{|CF_l|}{\sum_{l' \in L} |CF_{l'}|} \tag{3.29}$$

Since these probabilities of l, in L_u are independent from each other, it is possible to calculate the $p(l|L_u)$, by the multiplication of the conditional probabilities of l, on the every single subset of $L_{u.}$ All of the conditional probabilities in Equation 3.20 will be calculated by using the formula in Equation 3.19. $p(l)$ will be computed by Equation 3.21, which represents a weighted average of time-aware check-in popularity for activity l at time t, and general popularity without considering temporal feature. The β coefficient keeps the balance between two parts and will be computed in parameter tuning section in the Chapter 5.

3.4 Unified Framework

The Unified Recommendation System ($URS_{u,t,l}$) in [11] has been composed of Time-Aware and Space-Aware approaches, which have been explained earlier in this chapter. The linear interpolation is employed to keep the balance between the two mentioned approaches as showed in Equation 3.30.

$$URS_{u,t,l} = \alpha\ TRS_{u,t,l} + (1 - \alpha)SRS_{u,t,l} \tag{3.30}$$

Since the range of the scores calculated by each method is different, they should be normalized in order to be ready for combining with each other.

3.4.1 Extended Cascade Model

Another way of mixing both Time-Aware approach and the Space-Aware approach is cascading-based technique which incorporates the scores produced by one of the approaches by multiplying while producing the scores of the other one as an inter-coefficient. As remarked in Chapter 2, it is one of the regular techniques that is used for hybrid recommendations. For a given user u, and a given time slot t, while producing the scores for the location l by the ATRS method, the $Cascade\ ATRS_{u,t,l}$ can be calculated by multiplication of $SRS_{u,t,l}$ scores to the temporary $ATRS_{u,t,l}$, value and sort the rankings for performing the normal procedure of the recommendation. The same algorithm is employed for the $CascadeSRS_{u,t,l}$, with the difference that prepared final $ATRS_{u,t,l}$ scores used as the coefficient in the middle of the computation of $CascadeSRS_{u,t,l}$ scores.

CHAPTER 4

ACTIVITY-BASED RECOMMENDATION SYSTEM

In this chapter, our main contribution is described in detail. This chapter consists of two main sections. In the first section, concerning the studies related to the effect of different temporal features on the user check-in behavior, we propose our model for the time-aware activity-based recommendation system. In the second section an activity-correlated type of recommendation model is offered. It is adopted by collaborating the ideas of Space-Aware Recommendation System, which is introduced in Chapter 3, and a method of calculating correlations between activities.

As mentioned earlier, this work employs the ideas in [11], in other words, it can be considered as the foundation of this work. However, there are differences, improvements and innovations in comparison to our base work. The main difference is that work proposes a location-based recommendation system; whereas our system recommends the activities. To actualize this idea we combined the Gowalla data set to the Foursqaure data inventively to afford a suitable data set for this framework by providing the category types for the existing locations in Gowalla. Among the 9 general top-categories available in Foursquare, we selected 4 of them as shown in Figure 1 to work with their related sub-categories, so our Gowalla data set also was filtered and limited to cover selected sub-categories. This reasonable selection made us capable to recommend users such kind of entertaining and fun groups of activities.

Further, we extend the time concept as incorporating the type of the day to our framework. Considering the day in one side and how to utilize it with the hour parameter together in another side could be constituted as the contributions of this work. Furthermore, we proffer a coefficient in order to refine the effect of time better in our recommendation system by emphasizing the correlation more than the previous similar studies in this field.

Lastly, we leverage an existing statistical technique to calculate the correlation between different activities to be able to propose our activity-correlated recommendation framework.

Table 2. Selected top-categories and related sub-categories from Foursquare

Top Category	Sub Category
FOOD	Afghan Restaurant, African Restaurant, American Restaurant, Arepa Restaurant, Argentinian Restaurant, Asian Restaurant, Australian Restaurant, Austrian Restaurant, BBQ Joint, Bagel Shop, Bakery, Belarusian Restaurant, Belgian Restaurant, Bistro, Brazilian Restaurant, Breakfast Spot, Bubble Tea Shop, Buffet, Burger Joint, Burrito Place, Cafeteria, Café, Cajun / Creole Restaurant, Cambodian Restaurant, …
Arts And Entertainment	Aquarium, Arcade, Art Gallery, Bowling Alley, Casino, Circus, Comedy Club, Concert Hall, Country Dance Club, Disc Golf, General Entertainment, Go Kart Track, Historic Site, Laser Tag, Mini Golf, Movie Theater, Museum, Music Venue, Outdoor Sculpture, Performing Arts Venue, Pool Hall, Public Art, Racetrack, Roller Rink, Art, …
Nightlife Spot	Bar, Beach Bar, Beer Garden, Brewery, Champagne Bar, Cocktail Bar, Dive Bar, Gay Bar, Hookah Bar, Hotel Bar, Karaoke Bar, Lounge, Night Market, Nightclub, Other Nightlife, Pub, Sake Bar, Speakeasy, Sports Bar, Strip Club, Whisky Bar, Wine Bar
Event	Conference, Convention, Festival, Music Festival, Other Event, Parade, Stoop Sale, Street Fair

4.1 Time-Aware Activity-Based Recommender

As discussed in the previous chapter, although Basic Recommendation System which is originally the same as common collaborative filtering recommendation method, still is popular, there are certain features that could be incorporated in order to improve the performance. Time is one of the available options for our data which gives us extra information specifically in location recommendation field. Accordingly, in this section, we first study the influence of time on the user behavior. Then we aim to propose methods considering the different time-related features such as hour of the day or the day of the week in order to improve the performance of the baseline approach. Finally, we offer a unified time-inclusive framework.

4.1.1 Fundamental Investigations on the Data

As shown in the Figure 2, there is a strong correlation between the hours of the day and the check-in frequencies for the activities. For example, in the morning around 7:00 till noon the check-in frequencies for all of the top 10 frequent sub-categories decreased, because most of the people are attending to work or school in that period. On the other side, the highest check-in values are normally related to midnight or afternoon. It’s so logical, as the selected activities belong to categories that should be done in the leisure time.

Figure 3 also demonstrates an anticipatory pattern. In all curves, for all sub-categories, the maximum check-in values belong to the day Saturday or in some curves, to Sunday. It means that users have more chance to go to restaurants or have fun at weekend rather than on weekdays, which it has consistency with the real life as well.

These studies may convey trivial simple facts. But we can conclude irrefutable role of the temporal features in the user behavior. It is so clear that the check-in frequencies or in general, the pattern of users’ check-in, strongly depends to the time parameter and the moment the activity is done on that.

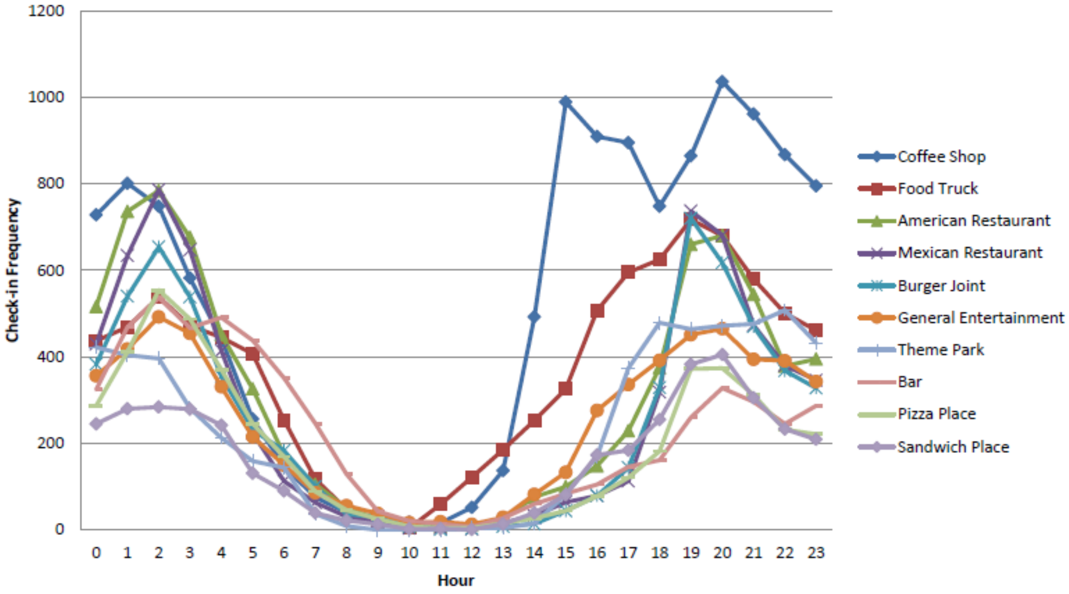

Figure 1. Correlation between hours of the day and check-in frequencies for the top 10 frequent sub-categories

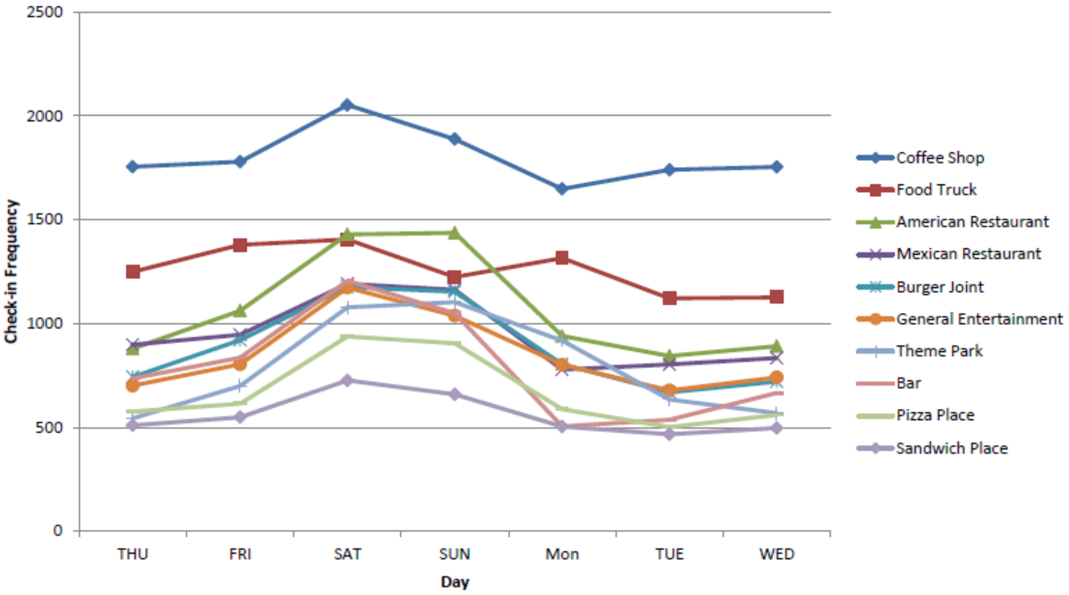

Figure 2. Correlation between day of the week and check-in frequencies for the top 10 frequent sub-categories

According to the literature survey we have conducted, there is only a few studies that investigate the effect of time through the recommendation process. Specifically, there is a perceptible deficiency in activity recommendation works. In the rest of this section, we show how we avail the approach in [11], presented in the Chapter 3, to elevate the activity recommendation system.

4.1.2 Time-Aware Model

After emphasizing on the undisputed role of time on the recommendation enhancement, we opted to increase the impact of time, with the hope that the performance of the presented temporal-based RS mentioned in the Chapter 3, increases. By using activity features instead of location, and also integrating the $\Upsilon_{t,a}$ parameter, we refined the formula in Equation 3.25 to proffer our activity-based model, Modified Time-Aware Recommendation Score ($MTRS_{u,t,a}$), for a given user u, at the time slot *t*, for the activity a, which is formalized in Equation 4.1.

$$MTRS_{u,t,a} = \frac{\sum_{v}(ATS_{u,v}\sum_{t'}\tilde{c}_{v,t',a}\cdot\rho_{t,t'})}{\sum_{v}ATS_{u,v}}\Upsilon_{t,a} \tag{4.1}$$

The $ATS_{u,v}$, $\tilde{c}_{v,t',a}$, and $\rho_{t,t'}$ factors are the Advanced Time-Aware Similarity, the smoothed check-in data, and the time slot similarity, whose formal definitions have been given in Equations 3.24, 3.23 and 3.22, respectively.

In order to calculate the $\Upsilon_{t,a}$ parameter, for each activity, the number of check-ins for activity a is counted with respect to a specific time *t*. All the values should then be normalized. In this way, the important correlation between time and activity posited and applied to the method. There is a simple idea behind this extension and general wisdom. In the real life, there is a general routine pattern for the entertainment-related activities or food-related ones with respect to the check-in time. As an example the most crowded period for "Night Clubs" is at night or midnight, and obviously it cannot be morning or noon. Likewise for "Bowling Saloons", they usually are crowded from the afternoon till night. In contrast, "Museums" normally are closed in the afternoon, hence the most frequent check-ins for them occurs before the hour they close.

Therefore, the mentioned parameter helps to reduce the unbiased recommendations by fading the effect of rare, uncommon or even noisy data. In a case that we produce top N recommendations for user x, for the time slot 10:00 am to 11:00 am, if one of the suggestions is "Night Bar", by using the $\Upsilon_{t,a}$ coefficient, we apply the general trend to measure the degree of validity of this suggestion. And for this example, since it’s not a habitual recommendation, its score will be decreased and accordingly its rank changes to a lower one in the recommendation results.

4.1.3 Unified Time-Aware Framework

In the previous section we tendered an extended version of a temporal based recommendation system which can be customized for whether using days as the time concept or hours. As we showed in the Figure 3, what day of the week can be as effective and valuable as the time of the day signifies. Henceforth, we decided to deploy our method not only for hourly time slots, but also for daily time slots.

We moved one more step forward, trying to study the effect of the combination of time and day together on the check-in frequencies trend. With this in mind that, the user mobility pattern not only depends on the hour or the day, but also it depends on the hour of a specific day. Hour and day concepts together create a new temporal concept.

Figures 4 and 5 describe the idea better. These figures have been plotted for the top 10 popular sub-categories that show the check-in frequency trend per hour separated by day, Figure 4 shows Saturdays pattern and Figure 5 illustrates Tuesday’s trend. These two days were chosen deliberately, since Saturdays are so crowded in respect of leisure activities check-ins and Tuesdays in contrast, known as normal working days. Almost all sub-categories have different patterns in each of these two days, but as significant ones, "Bar" and "Coffee Shop" are appropriate examples obviously. As can be seen, considering both temporal concepts can lead us to more accurate recommendations, closer to real life experiences.

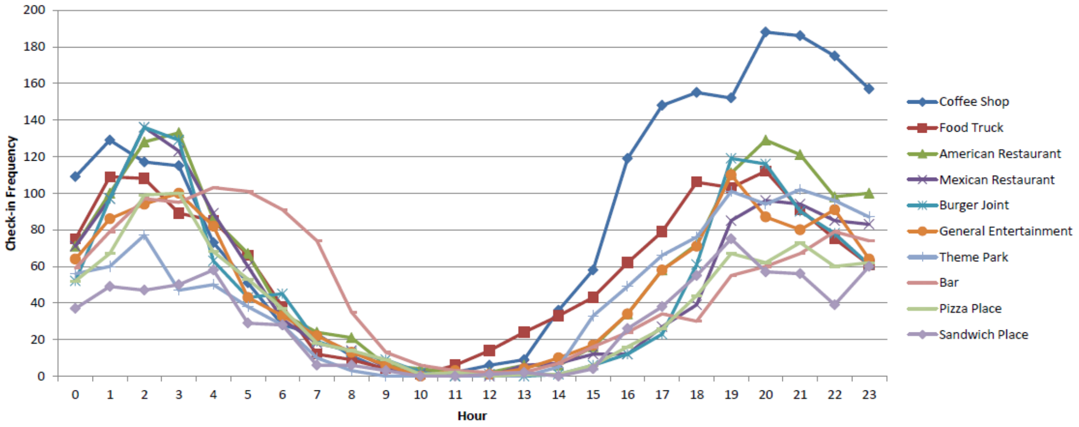

Figure 3. Check-in frequencies pattern for 24 hours of Saturdays for the top 10 frequent sub-categories

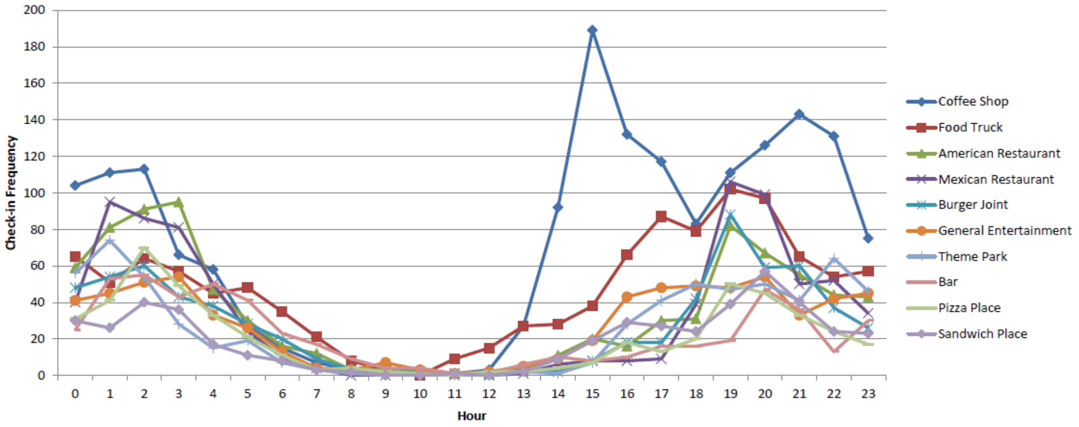

Figure 4. Check-in frequencies pattern for 24 hours of Tuesdays for the top 10 frequent sub-categories

Therefore, we employ our finalized version of the time-aware recommendation method, once for hour-based data which we call it Hour-Based Recommendation System ($MTRS_{u,t,a}^{HB}$), and also once for the day-based, called Day-Based Recommendation System ($MTRS_{u,t,a}^{DB}$), respectively. Then we combined these two approaches creatively as in Equation 4.2.

$$UTRS_{u,d,h,a} = MTRS_{u,t,a}^{HB} + MTRS_{u,t,a}^{DB} \tag{4.2}$$

As it can be seen in the formula, the Unified Time-Aware Recommendation System ($UTRS_{u,d,h,a}$) is a model for calculating the activity a score for a given user u, in a given day d, at a specific hour h. Since the scores of the two models, $MTRS_{u,t,a}^{HB}$ and $TRS_{u,t,a}^{DB}$ are in different scales, in order to be able to merge them, we first normalize the scores of each one individually. The top highest ranked items will then be merged together for the purpose of providing new scores for the Unified Time-Aware Recommendation System.

4.2 Activity Correlation-Aware Approach

In this section we propose another activity-based recommendation method from a different point of view by focusing on the correlation among items solely. Based on the analyses performed in several works such as [11] [9] [42] user check-in pattern seems to be affected by the geographical closeness of the starting point and the target. In other words, the more a POI is far from the current situation of the user, the less it is probable to be selected as a target.

Correspondingly, this observation inspired us to use the relationships among sub-categories. At the first glance the idea may seem indeterminate. Particularly, in the mentioned works the distance between locations is directed which is one of the available features in the data in the form of longitude and latitude. But the main challenge here is how to measure the relationships between sub-categories in order to be able to making decisions about them. In the following sections we will introduce our method and the idea we applied to address the mentioned problem.

4.2.1 Activity-Correlated Model

The key challenge in this phase as implied above is finding a method to make us capable of measuring the distance between two sub-categories. The word distance here can be interpreted as correlation or a similarity measure between each pair of activities.

One simple approach for having such a measure could be as follows: assume the top-categories and sub-categories as a graph in a way that sub-categories are leaves of the graph, while jumping from one sub-category to another which belongs to the different top-category, the weight is twice the weight of internal moves between the nodes of a same top-category. But the problem with this approach is that for the transitions between nodes belong to the same top-category has no special effect. So each time we will waive a remarkable amount of activities for recommendation calculation.

Due to this drawback, instead we deploy a logical simple idea. The idea is based on building queries by combining every two single activities; passing them on the Google and take the result counts. In Table 2, a sample list of these weights is presented. Since they have different ranges and belong to different samples we normalize them to have the same range of values in order to guarantee robustness.

Table 3. Sample of activity correlation weights

Sub-category	Sub-category	Normalized Weight
Stadium	Pizza Place	0.000230932
Stadium	French Restaurant	0.000137712
Stadium	Diner	0.006875
Stadium	Sports Bar	0.000948093
Stadium	Sushi Restaurant	0.000170551
Stadium	New American Restaurant	8.17797E-05
Pub	Indonesian Restaurant	0.000508475
Pub	Whisky Bar	0.000332627
Pub	Nightlife Spot	0.000111229
Pub	Mongolian Restaurant	2.30932E-05
Pub	Creperie	0.005275424
Pub	Middle Eastern Restaurant	9.18432E-05
Art Gallery	Bakery	0.010391949
Art Gallery	Coffee Shop	0.001058263
Art Gallery	BBQ Joint	0.000210805
Art Gallery	Chinese Restaurant	0.000413136
Art Gallery	Performing Arts Venue	4.67161E-05
Art Gallery	Stadium	0.029661017

It is possible to interpret these weights as the amount of user interest for switching from one activity to another one. Assuming that if a user is doing the activity A_i, the probability that the A_j, might be a candidate as a next successive activity follows Equation 4.3.

$$p(A_j/A_i) = \frac{Correlation\ (A_i, A_j)}{\sum_{A_t \in A_u, A_t \neq A_i} Correlation\ (A_i, A_t)} \tag{4.3}$$

Now based on the history of the user u, and the activity history set A_u, utilizing the naïve Bayesian method we can compute the Activity Correlation-Aware Recommendation Score ($ACRS_{u,a}$), for a new activity, a.

$$ACRS_{u,a} = p(a|A_u) = \ p(a) \prod_{a' \in A_u} p(a'|a) \tag{4.4}$$

$$p(a) = \frac{|CF_a|}{\sum_{a' \in A} |CF_{a'}|} \tag{4.5}$$

Since these probabilities of a, in A_u are independent from each other, it is possible to calculate the $p(a|A_u)$, by the multiplication of the conditional probabilities of a, on the every single subset of A_u. All of the conditional probabilities in Equation 4.4 are calculated by the Equation 4.3 formula. The *p(a)* also is the popularity of activity *a* through all the other activities which is formulated by the Equation 4.5.

$$TACRS_{u,a} = p(a|A_u) = \ p_t(a) \prod_{a' \in A_u} p(a'|a) \tag{4.6}$$

$$p_t(a) = \beta \ \frac{|CF_{a,t}|}{\sum_{a' \in A} |CF_{a',t}|} + (1 - \beta) \ \frac{|CF_a|}{\sum_{a' \in A} |CF_{a'}|} \tag{4.7}$$

The Equation 4.6, Time-Aware Activity-Correlated Recommendation System presents the time-aware version of the Equation 4.4 by incorporating temporal feature in $p_t(a)$. $p_t(a)$ will be computed by the Equation 4.7 which represents the weighted average of time-aware check-in popularity for activity a at time t, and general popularity without considering temporal feature. The β parameter keeps the balance and will be computed in parameter tuning section in the next chapter.

Equation 4.6 is the same with the one that is used in the mentioned work, but for better performance, we devised Equation 4.8 which is more accurate with respect to accuracy under $\Upsilon_{t,a}$ parameter that is introduced in the second section elaborately.

$$MACRS_{u,a} = p(a|A_u) = \Upsilon_{t,a} \times p(a) \prod_{a' \in A_u} p(a'|a) \tag{4.8}$$

4.2.2 Unified Activity-Correlated Framework

Similar to the sub-section 4.1.3 we implement the Modified Activity-Correlated Recommendation method, for hourly-based data which we call it Hour-Based Modified Activity-Correlated Recommendation System ($MACRS_{u,t,a}^{HB}$), and also for the day-based, called Day-Based Modified Activity-Correlated Recommendation System ($MACRS_{u,t,a}^{DB}$), respectively. Then we merge these two ideas as shown in the equation (4.9).

$$UACRS_{u,d,h,a} = MACRS_{u,t,a}^{HB} + MACRS_{u,t,a}^{DB} \tag{4.9}$$

The Unified Activity-Correlated Recommendation System ($UACRS_{u,d,h,a}$) is a model for calculating the activity a score for a given user u, in a given day d, at a specific hour h. Since the scores of the two models, $MACRS_{u,t,a}^{HB}$ and $MACRS_{u,t,a}^{DB}$ are in different scales, in order to be able to merge them, we first normalize the scores of each one individually. The top highest ranked items will then be merged together for the purpose of providing new scores for the Unified Activity-Correlated Recommendation System.

CHAPTER 5

EVALUATION AND EXPERIMENTAL RESULTS

In this chapter the performance of the methods discussed in Chapter 3 and Chapter 4, is evaluated. Firstly, we introduce our dataset and the characteristics of it. Then the evaluation metrics used in this work are specified. Next, the results for all of the methods proposed in this work are demonstrated besides the comparison with the baseline. Finally, the parameter optimization analysis is presented with graphical charts.

It may be useful to indicate that, all algorithms have been implemented in c# and have been performed on a machine with 4 GB ram and 2.60 GHz, Intel Core i5 CPU.

5.1 Data Set

As mentioned before, Gowalla is one of the famous data sets for location-based social networks. Similar to many other works [33] [31] [43] [44] in this work also we take advantage of this popular data set. Gowalla was an active web site through 2009-2012 for collecting check-ins made by mobile device users [42] [45]. It contains 6,442,890 check-ins for 196,591 users. Each record of data as shown in the Table 3, presents single check-in information which includes user ID, timestamp, location coordinates which are latitude and longitude, and location ID.

Table 4. A snapshot of the Gowalla dataset

User ID	Check-in Timestamp	Latitude	Longitude	Location ID
8921	2010-09-26 01:53:55	37.6184424117	-122.0361059904	874428
8921	2010-09-26 01:36:21	37.606786695	-122.0626224146	397511
8921	2010-09-24 04:33:50	37.6184424117	-122.0361059904	874428
8921	2010-09-24 03:30:54	37.606786695	-122.0626224146	397511
8921	2010-09-22 04:22:24	37.560765	-122.0113623667	4188291
8921	2010-09-22 03:38:19	37.5437036333	-121.9856487667	151344
8921	2010-09-21 03:26:05	37.550501817	-121.9798415	1209961
8921	2010-09-21 00:44:32	37.5669922255	-122.0519900322	977314
8921	2010-09-20 00:19:22	37.619043	-122.0351608833	422113
8921	2010-09-19 23:11:28	37.6448693654	-122.0619263179	712806
8921	2010-09-18 19:16:06	37.5669922255	-122.0519900322	977314
8921	2010-09-13 04:38:47	37.60796145	-122.0660003333	299894
8921	2010-09-13 01:40:54	37.3932797198	-122.0791634917	23599
8921	2010-09-13 00:00:24	37.3940297167	-122.07971555	3608646
8921	2010-09-12 23:49:11	37.3878924161	-122.0828863978	2664667
8921	2010-09-11 04:59:05	37.59757705	-122.0699151167	352750

Gowalla data set is publicly available and it covers a very fast area geographically, however, there is some missing information in the data set. One of them is the lack of having direct information about geographical locations. Each coordinate contains longitude and latitude to point to an exact location, but we need to have the fields related to countries and cities beforehand in order to decide which area or areas we should focus on. As the other deficiency, there is no field related to the venues of the locations which directly indicates the type of activity that can be done there. For example, it can be a "Chinese Restaurant" or a "Bowling Saloon" as well. So, having location categories is a requisite for a venue type recommendation system.

To deal with the first issue we implemented Google Reverse Gecoding. As input, it takes coordinates and for output it gives addresses fetched by Google Maps[7]. It might be also useful for the future studies related to Gowalla since the complete address for each location is at hand.

Among 72 countries we opted not only to work on just cities from one country but also the cities which are near to each other. Seeing that geographical situation plays an important role in the habits and cultures of the people from that region, we want to have a homogeneous mixture of people as much as possible. Finally, Nevada and California were choices. On one side, they are the states near each other with active people who provide sufficient check-ins. On the other side this selection makes the comparison with our base paper [11] much more meaningful since the states are the same.

For the second issue we utilized Foursquare API which takes check-in information as input and generates the information related to the category hierarchy of the location and the name of the place, such as Burger King. The hierarchy has at most 3 levels. For the first level, which is the most abstract one has 9 top categories which are: arts and entertainment, college and university, event, food, nightlife spot, outdoors and recreation, professional and other places, residence, shop and service, travel and transport.

For our activity-based recommendation system we decided to focus on sub-categories, since top-categories are too abstract and sub-sub-categories are also not available for some sub-categories and in addition, they are too detailed. Among sub-

[7] https://www.google.com/maps

categories some of them are not suitable for recommendation. For example, recommending to a user to go to "Post Office" or "Medical Center" or "Bank" or "Home" at a specific time in a specific day in the week may seem meaningless. For this reason except arts and entertainment, event, food, nightlife spot other top-categories have been omitted. Then the users who checked-in fewer than 4 distinct activities were eliminated.

For our location-based recommendation system we performed another filtering in way that the users who checked-in fewer than 5 distinct locations and also locations with fewer than 5 user check-ins were eliminated.

After data preprocessing, data filtering, and excluding Nulls (for some locations the venue category is Null) we have two data sets, one for the location-based experiments containing 176741 check-ins made by 10008 users at 9414 distinct locations and one for the activity-based experiments containing 168640 check-ins made by 7220 users belonging to 151 different sub-categories. The complete statistics for the two prepared data sets are available in Table 4.

Table 5. Data set statistics

	Location-Based Data	**Activity-Based Data**
Date Interval	2009-03-15 up to 2010-10-22	2009-03-15 up to 2010-10-22
No. of Users	10008	7220
No. of Items	9414	151
No. of Check-ins	176741	168640
Sparsity for Day-based Cube	99.79 %	98.35 %
Sparsity for Hour-based Cube	99.93 %	99.48 %

The snapshot of the new data set created by merging the Gowalla and Foursqaure data sets is shown in Table 5.

Table 6. A snapshot of the combined data set by Gowalla and Foursquare

User ID	Check-in Timestamp	Latitude	Longitude	Location ID	Sub Category	Day
54	2010-02-27 17:29:34	34.01786965	-118.48907148	113	Deli / Bodega	Sa
54	2010-02-17 01:20:05	34.0409025	-118.4435497	3571	Tea Room	We
54	2010-02-07 02:00:31	33.85709246	-118.290575879	5218	Food Truck	Su
54	2010-02-07 01:06:19	33.68807189	-117.83412784	232	Bakery	Su
54	2010-01-31 19:53:55	34.06243882	-118.443533026	8146	Food Truck	Su
54	2010-01-31 17:29:44	34.0996449	-118.327750283	328	Pub	Su
54	2010-01-26 03:30:22	34.0409025	-118.4435497	3571	Tea Room	Tu
54	2010-01-24 01:43:19	36.12451275	-115.169570446	1764	Lounge	Su
54	2010-01-23 00:46:35	34.06361007	-118.22408080	28	Wine Bar	Sa
54	2010-01-22 02:21:42	34.04048323	-118.427724833	973	Burger Joint	Fr

5.2 Evaluation Metrics

One of the most popular and standard metrics for testing the performance of the recommendation systems is Precision and Recall pair. Precision can be defined as the fraction of retrieved recommendations which are relevant as well and Recall can be defined as the fraction of relevant recommendations that are retrieved. Before presenting the formula for Precision and Recall there are some notations which needs to be clarified [46]:

True Positive (TP): The number of relevant items in the recommendation list or the items which are presented as the result of recommendation that have been chosen by the user before.

False Positive (FP): The number of items in the recommendation list, which are retrieved wrongly and are not related to the user.

True Negative (TN): The number of items which are relevant to the user at the specific time but could not be retrieved.

False Negative (FN): The number of items which are not relevant and also are not retrieved.

The formalization of the Precision and Recall for all the users in a given time *t*, is shown in Equation 5.1 and 5.2.

$$Precision_t = \frac{\sum_{u \in U} \mathrm{TP}_{u,t}}{\sum_{u \in U}(\mathrm{TP}_{u,t} + \mathrm{FP}_{u,t})} \tag{5.1}$$

$$Recall_t = \frac{\sum_{u \in U} \mathrm{TP}_{u,t}}{\sum_{u \in U}(\mathrm{TP}_{u,t} + \mathrm{TN}_{u,t})} \tag{5.2}$$

For a brief example, in a case that our model produces 10 recommendations and the test set contains 20 items related to a specific user and time, if there are 3 relevant or common items among 10, in the recommendation list, the $Precision_t$, would be 3/10 and the $Recall_t$, would be 3/20, respectively. The overall Precision and Recall are calculated by the average of the Precision and Recall values in Equation 5.1 and 5.2 over all time slots.

Since in some of the methods it is possible to have a significant improve for Precision and at the same time a drop for Recall, we need a metric to give a general information gained by the overall trend of Precision and Recall. For this reason, to evaluate the performance of our proposed models, we choose F-score as one of the well-known metrics in recommendation field. F-score or F-measure has been defined as the harmonic mean of Precision and Recall [47]. The general formalization of the F-score defined in Equation 5.3.

$$F_{\beta} = (1 + \beta^2)\frac{Precision\,.\,Recall}{(\beta^2\,.\,Precision) + Recall} \tag{5.3}$$

Coefficient β which is a positive real value controls the weight of the importance of the Precision and Recall. The common version of this formula that is applied in this section, uses the 1 as the value for β, which is called balanced F-score or F_1 score. F_1 score gives the same weight to Recall and Precision. F_2 score and $F_{0.5}$ score are two other common measures which the first one gives the Recall more importance and the second one gives more value to the Precision respectively.

The Precisions and Recalls and accordingly F-scores normally are calculated for different recommendation set sizes which is commonly 5, 10, and 20. In our experiments we evaluate our metrics for N = 1,2,3,4,5 and 10, 15, 20. We are interested to see how the trend of the recommendation accuracy changes by the number of top suggestions returned by the system. Precision at N = 1, means that the model performance has been evaluated when just returns one single item as the top 1 recommendation. Furthermore, $F_{0.5}$ score also is computed for all of the experiments apart from F_1 score, in order to visualize the improvement of the Precision. In fact, we are interested the methods with higher Precision even if the Recall is not very improved, since top N recommendations relevancy is in the first priority.

5.3 Evaluation Method

For experiment testing, we used 4-fold cross validation method so we divided both our combined data sets into three non-overlapping sets as follows: 25% of the distinct locations/activities from each user history were selected randomly as the test set and 12% as development data dedicated for parameter tuning. The rest of the data was labeled as training data. Based on the history of users, the recommendation system estimates the score for each candidate items; the items not appeared in the user history. By these calculated scores items will be ranked and the top-highest N items will be recommended to the target user as the result of the system.

5.4 Experiments for PIP Similarity And NHSM Similarity Extensions

The performance of new methods for measuring similarity discussed in section 3.1.1 and section 3.1.2 as PIP similarity and NHSM similarity respectively, compared with the basic method that measures similarities by cosine similarity in Figure 6 and Figure 7. As we can see in here, the PIP similarity has a weak performance; because, first of all proportion of common locations between two users is not taken into account and second of all in this method the difference of users' preferences is not considered.

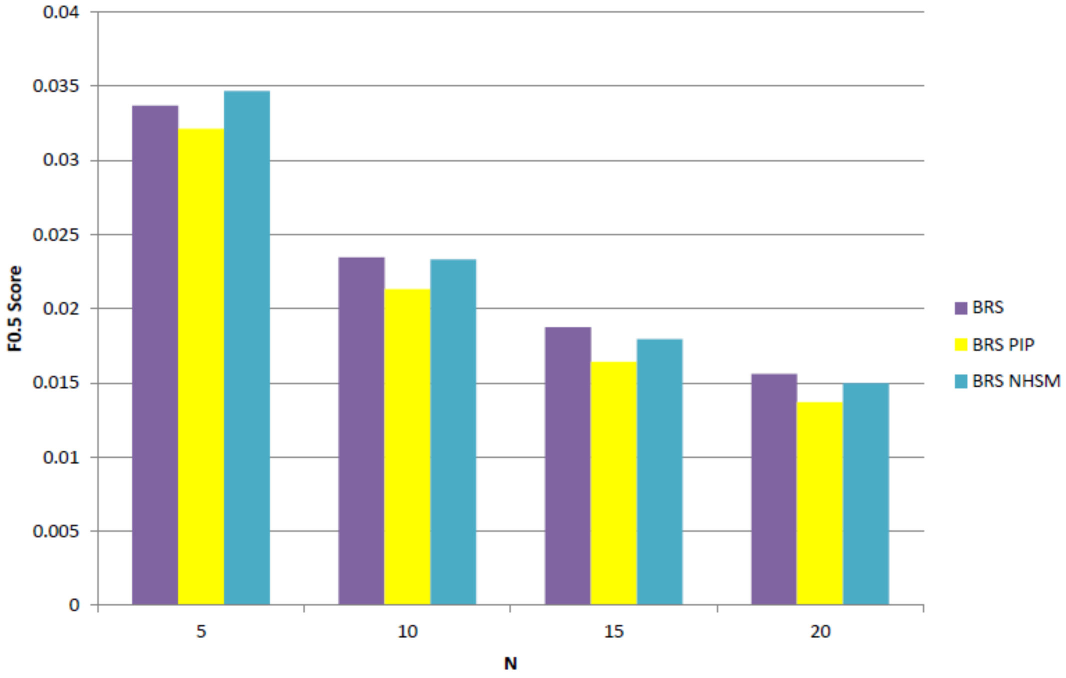

Figure 5. F0.5 score values for the methods *BRS_{PIP}*, and *BRS_{NHSM}*, vs. *BRS*

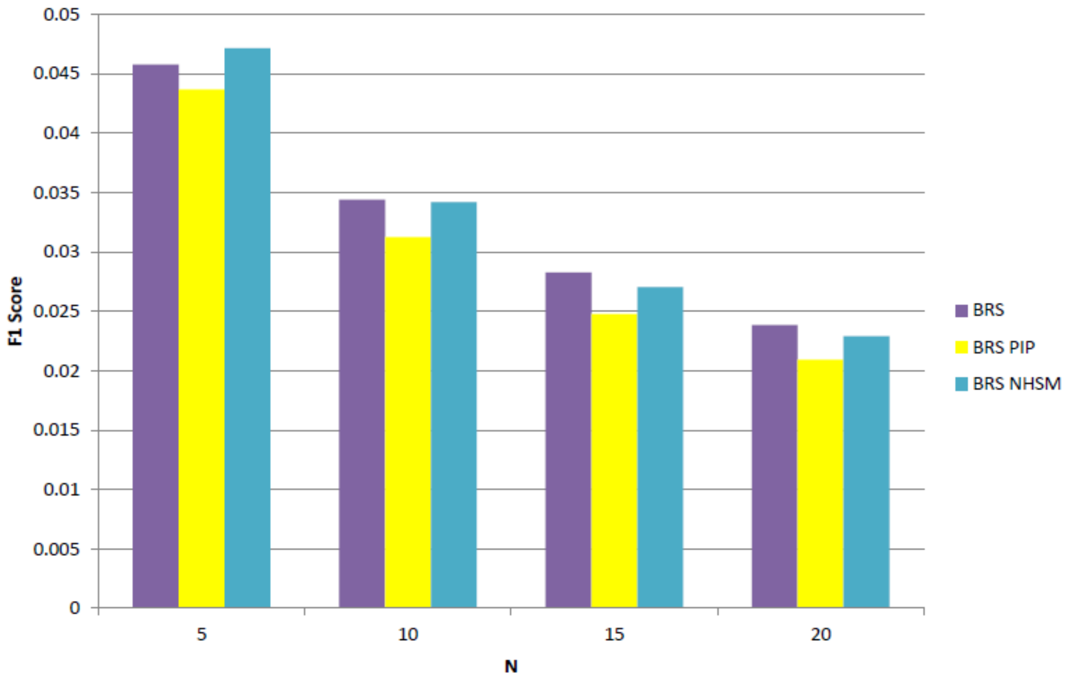

Figure 6. F1 score values for the methods BRS_{PIP}, and BRS_{NHSM}, vs. BRS

5.5 Experiments for Location-Based Unified Framework

The Location-based models which have been introduced in Chapter 3 are evaluated in this section. There are 5 techniques as, $ATRS_{u,t,l}$, $SRS_{u,t,l}$, $URS_{u,t,l}$, $Cascade\ ATRS_{u,t,l}$, $Cascade\ SRS_{u,t,l}$, whose performance in terms of accuracy is demonstrated in Figure 8 and Figure 9. All of these five techniques compute the recommendation score of a location l, for a given user u, at a specific hour-based time slot t. Then in each method, the scores produced for all locations is sorted in descending order, and top N is selected as the final result of recommendation system. It should be remarked that the value of β parameter used in $SRS_{u,t,l}$ method is 0.9 which is estimated by parameter optimization.

As shown in the figures, for the first few recommendations the Unified Recommendation System, which combines ATRS and SRS with linear interpolation outperforms the rest of the methods. But for the rest of the recommendations Cascade ATRS which uses the ATRS as the main model and inject the SRS scores as the inter-process scores, almost outperforms the URS. Because of applying multiplication for combining SRS and ATRS, the first few scores, especially the first one will be negatively affected by the low efficiency of SRS. As N increases, the effect of ATRS becomes more noticeable, till a balance between SRS and ATRS is made. It is clear that because of the weak performance of SRS, the Cascade SRS also has a poor performance.

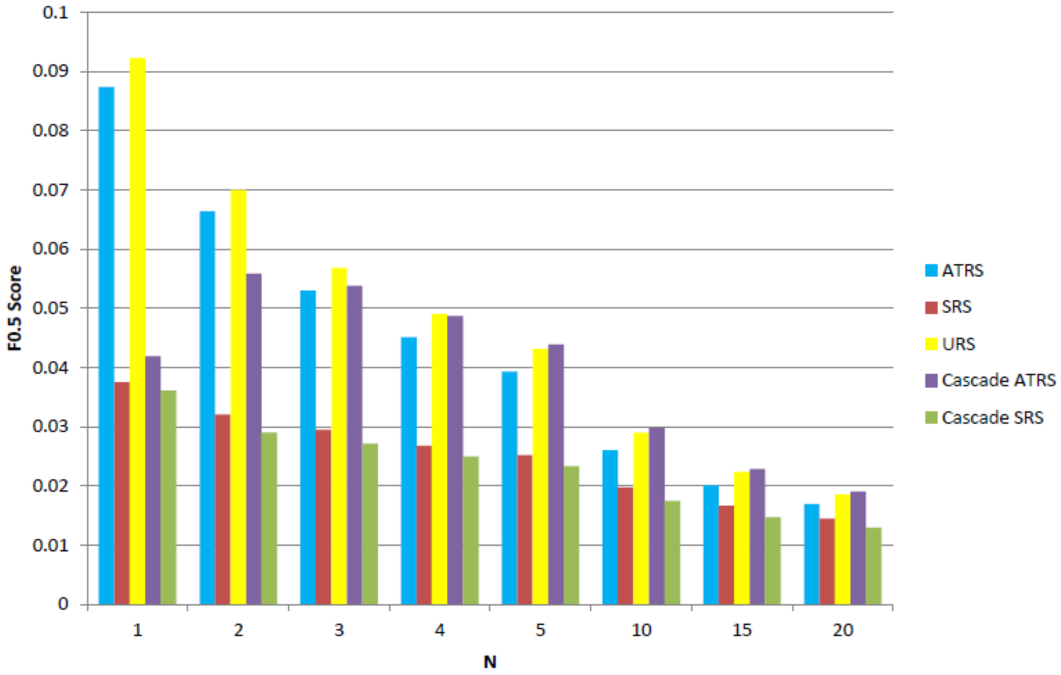

Figure 7. F0.5 score values for *URS, Cascade ATRS, Cascade SRS* vs. *SRS* and *ATRS*

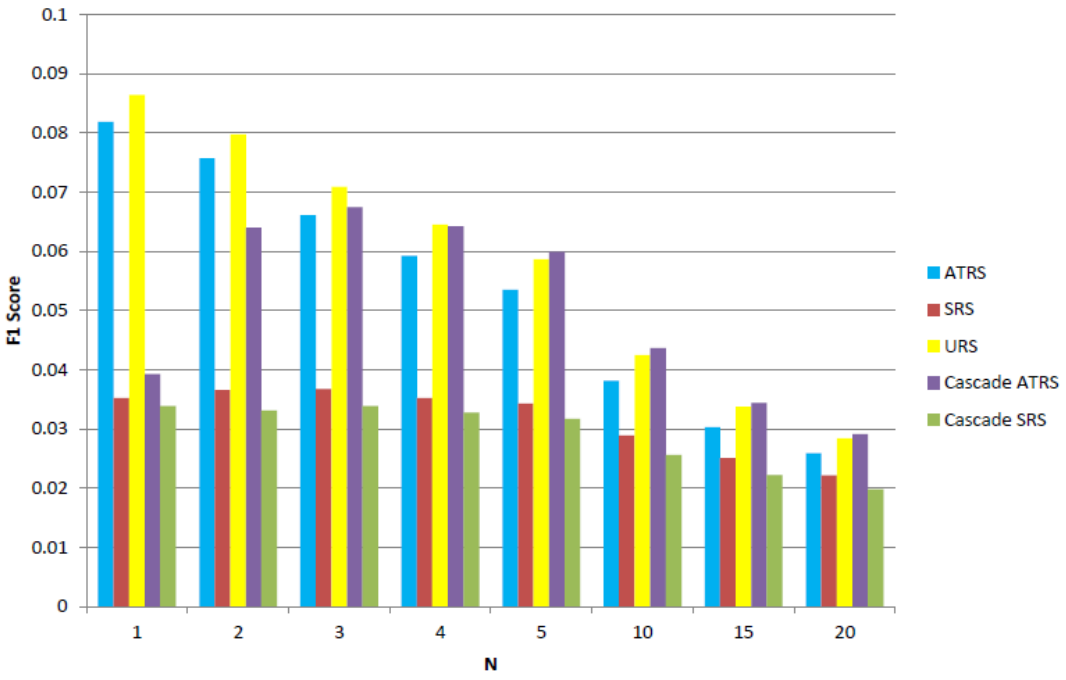

Figure 8. F1 score values for *URS*, *Cascade ATRS*, *Cascade SRS* vs. *SRS* and *ATRS*

5.6 Experiments for Activity-Based Models

In this section, the experiments related to activity-based approaches are presented. In the two first sub-sections, we discuss the efficiency of time-aware models based on the hourly time slots and the daily time slots respectively. In the third sub-section, the integration of the hour feature and day feature as the temporal information is analyzed. The next three sub-sections have the same order as the mentioned first three sub-sections regarding that they examine the activity-correlated model's performance. It is noteworthy that all of the upcoming models are assumed to acquire the top N activities, suggesting to the typical user *u*, and time slot *t*, which can be hour-based or day-based.

5.6.1 Performance of Hour-Based Time-Aware Models

The two first experiments depicted in Figure 10 and Figure 11, belong to hour-based approaches in the Time-Aware Activity-Based System. In Hour-based experiments, the time slots have been adjusted at hourly intervals; hence there are 24 time intervals. In the figures below, the two time-considered methods have been compared with the basic approach. As it was expected, both the Modified Time-Aware Recommendation System and the Time-Aware Recommendation System, outperformed the Basic Recommendation System whether in terms of F0.5 score or F1 score. By increasing the N the performance of all methods will be the same since the total number of activities is less.

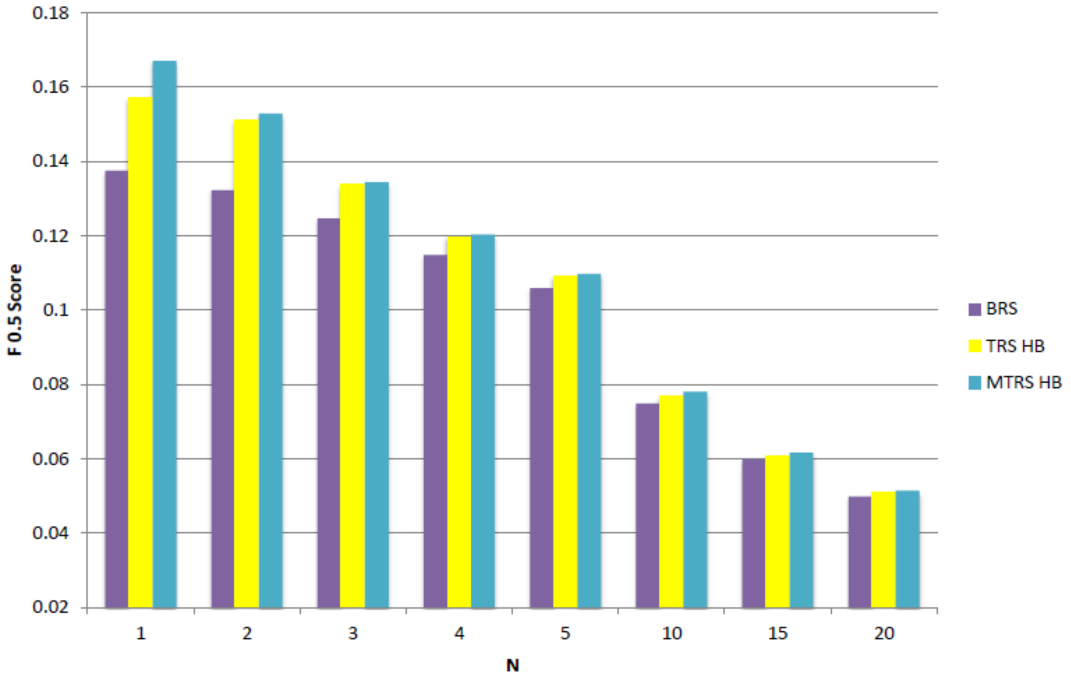

Figure 9. F0.5 score values for *TRS_{HB}*, and *$MTRS_{HB}$* vs. *BRS*

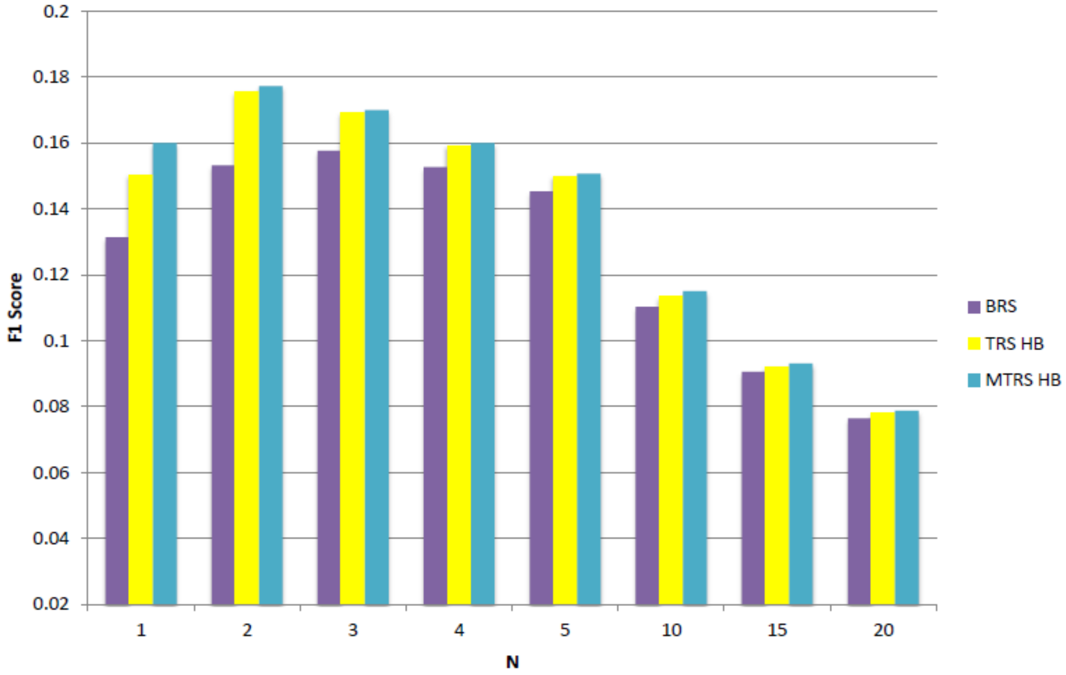

Figure 10. F1 score values for $\boldsymbol{TRS_{HB}}$, and $\boldsymbol{MTRS_{HB}}$ vs. $\boldsymbol{BRS}$

It should be emphasized that $MTRS_{HB}$ method successfully could outperform the TRS_{HB}, which means that using the $\Upsilon_{t,a}$ parameter has a promising influence on the method efficiency.

5.6.2 Performance of Day-Based Time-Aware Models

The next two figures (Figure 12 and Figure 13) illustrate day-based experiments in the Time-Aware Activity-Based System. For day-based experiments, there are 7 time slots, which have been adjusted by the days of a week. By increasing the length of time slot from an hour to a day, the sparsity is decreased as data become denser. Therefore, the precision will be improved. However, the poorer values for recall are observed due to the increased ground truths for each user by expanding the length of the time slots.

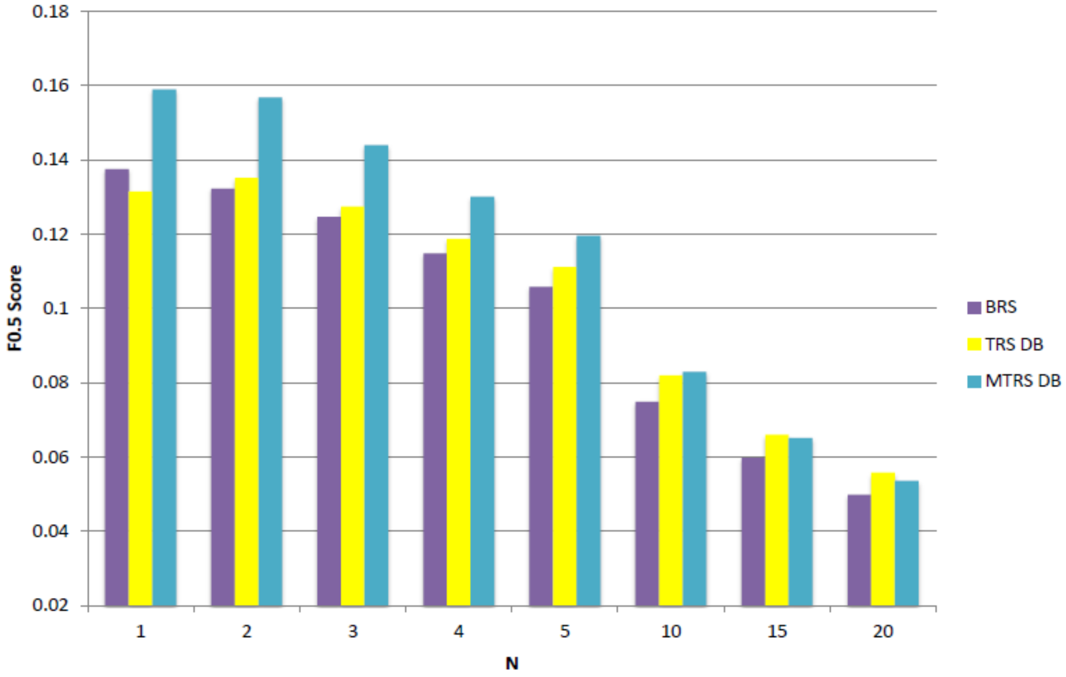

Figure 11. F0.5 score values for *TRS*$_{DB}$, and *MTRS*$_{DB}$ vs. *BRS*

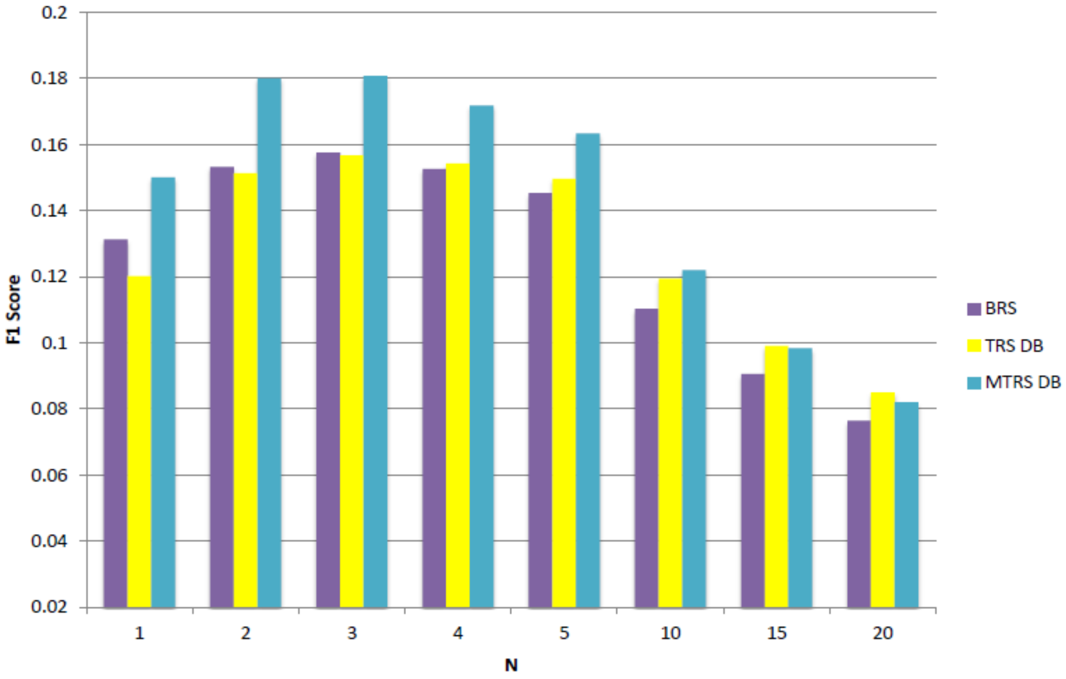

Figure 12. F1 score values for *TRS*$_{DB}$, and *MTRS*$_{DB}$ vs. *BRS*

5.6.3 Performance of Unified Time-Aware Model

In this section the charts for the unified model, is presented, which considers both day and hour at the same time as temporal features. As Figure 14 indicates a dramatic increase for the UTRS method for initial values of N, it means that the precision in the current model has been improved greatly rather than recall which is shown in Figure 15. Should be notified that precision is more significant since producing a few correct suggestions are highly demanded by the users rather than producing a lot of closer suggestions. In the reality also almost all the people are interested in the top few first recommendations.

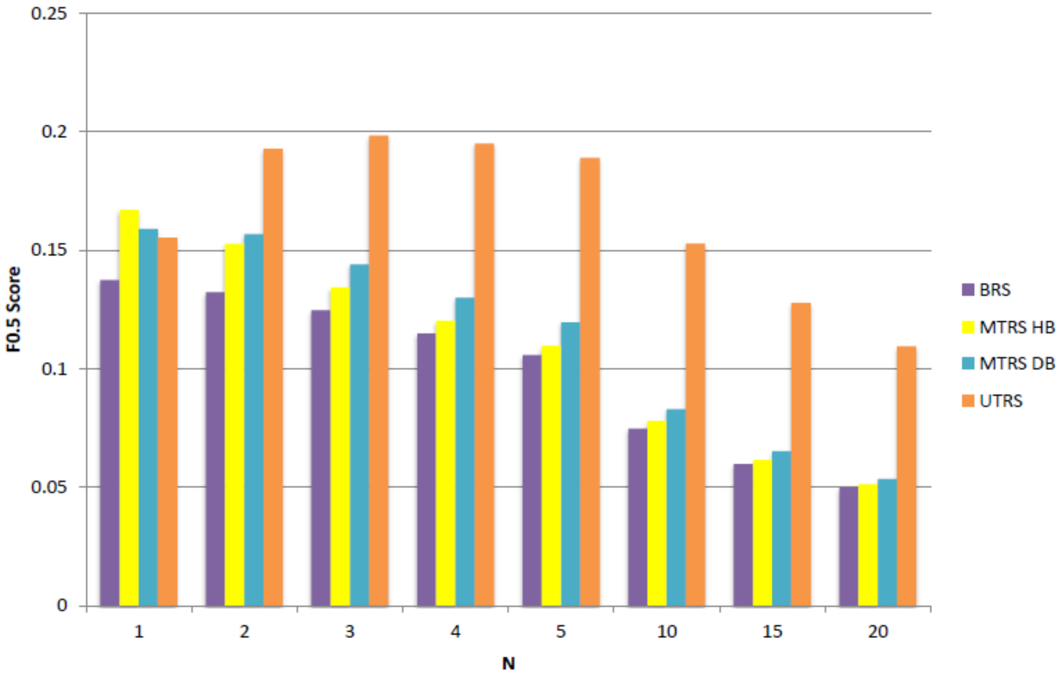

Figure 13. F0.5 score values for *UTRS* vs. *MTRS$_{HB}$* and *MTRS$_{DB}$* and *BRS*

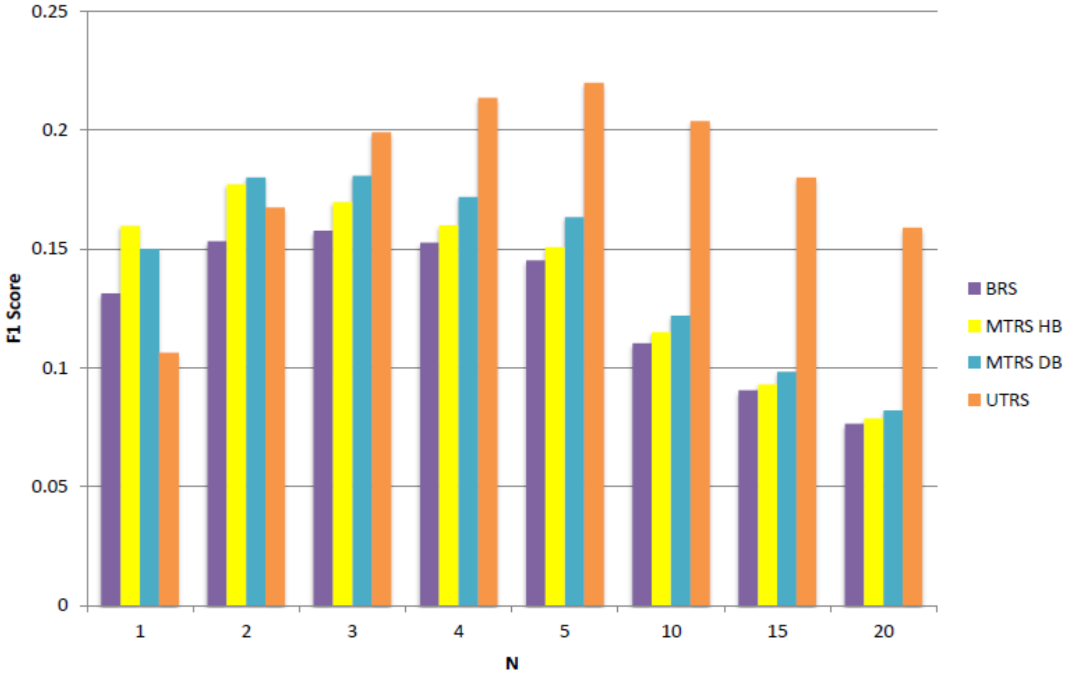

Figure 14. F1 score values for *UTRS* vs. *MTRS$_{HB}$* and *MTRS$_{DB}$* and *BRS*

It is noteworthy to point that, the average improvement percentage of UTRS in comparison to BRS, is around 75% for F0.5 score metric, and 50% for F1 score, which indicates a significant progress for the proposed method.

5.6.4 Performance Of Hour-Based Activity-Correlated Model

Figure 16 and Figure 17 show the performance of hourly-based approaches in the activity-correlated model evaluated by F0.5 score and F1 score. Incorporating the temporal features whether explicitly by the $p_t(a)$, as shown in Equation 4.7, or implicitly by the $\Upsilon_{t,a}$ parameter, for the TACRS$_{\mathrm{HB}}$ and MACRS$_{\mathrm{HB}}$ models causes significant improvements for almost all of the values of N. It should be remarked that the value of β parameter used in TACRS and MACRS methods, whether day-based or hour-based is 0.7 which is estimated by parameter optimization.

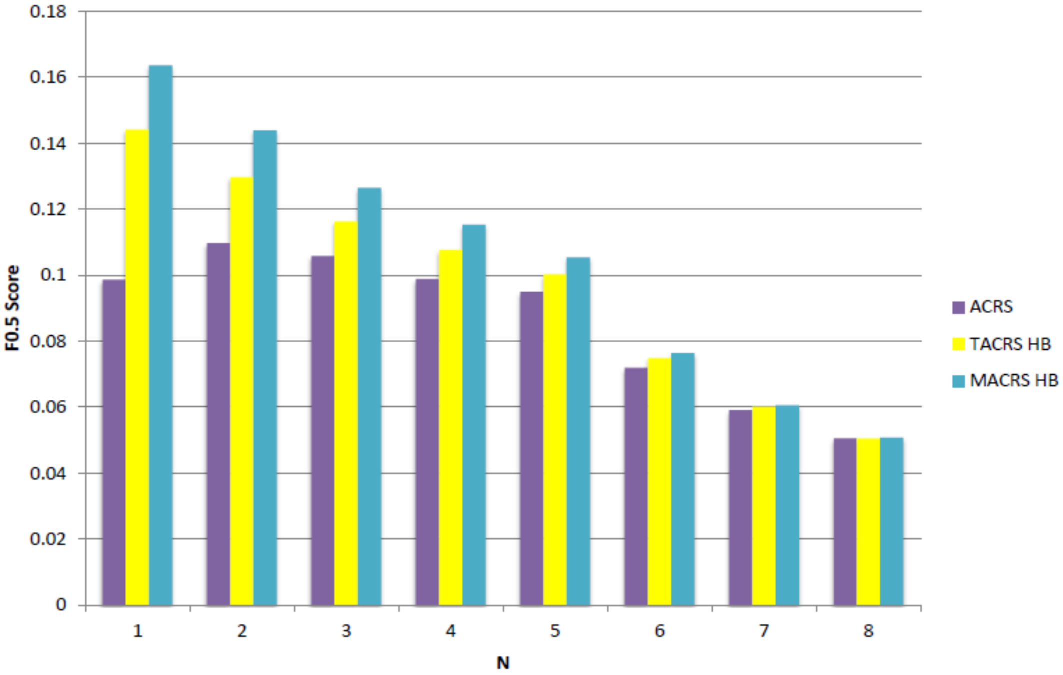

Figure 15. F0.5 score values for *TACRS*$_{HB}$, and *MACRS*$_{HB}$ vs. *ACRS*

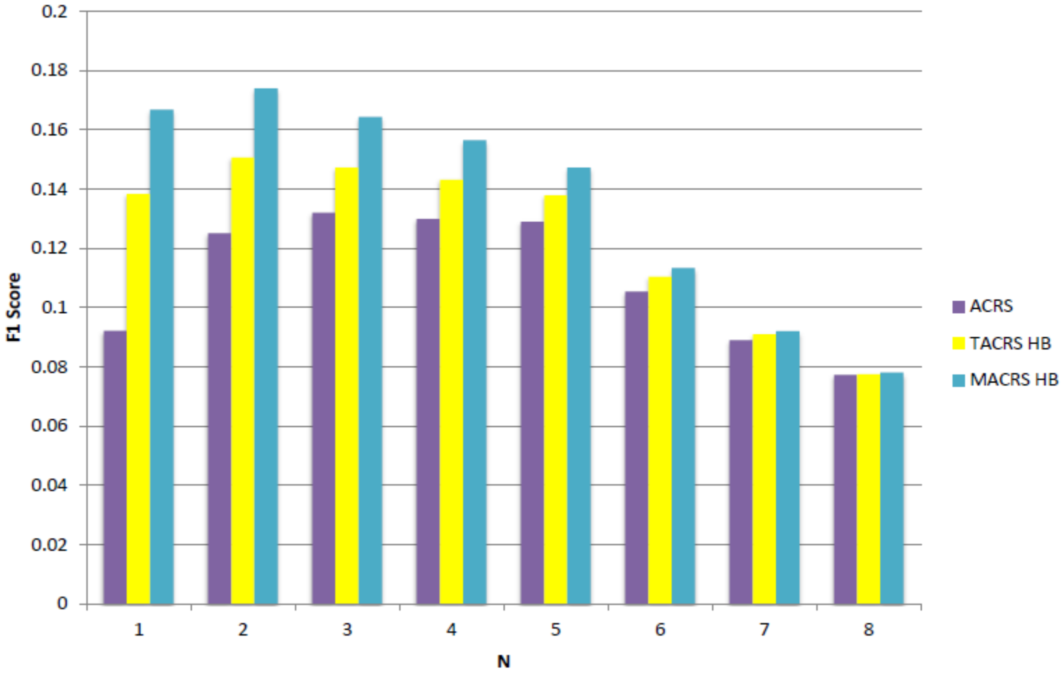

Figure 16. F1 score values for *TACRS*$_{HB}$, and *MACRS*$_{HB}$ vs. *ACRS*

5.6.5 Performance Of Day-Based Activity-Correlated Models

The result of the day-based activity-correlated models, TACRS and MACRS which are compared with the ACRS is visualized in the Figure 18 and Figure 19 used the metrics F0.5 score and F1 score respectively. It can be seen that the precision is more improved in comparison with recall since the F0.5 socre values show better performance for TACRS and MACRS rather than F1 score values. This improvement is occurred due to incorporating the temporal feature and the activity correlation feature as we discussed in the Chapter 4. Due to the negligible impact of time in the TACRS method, the performance in comparison with MACRS, for the top recommendations is weak. But by increasing the N, the TACRS reaches the MACRS due to have a fewer number of total items in this system which is the total number of activities.

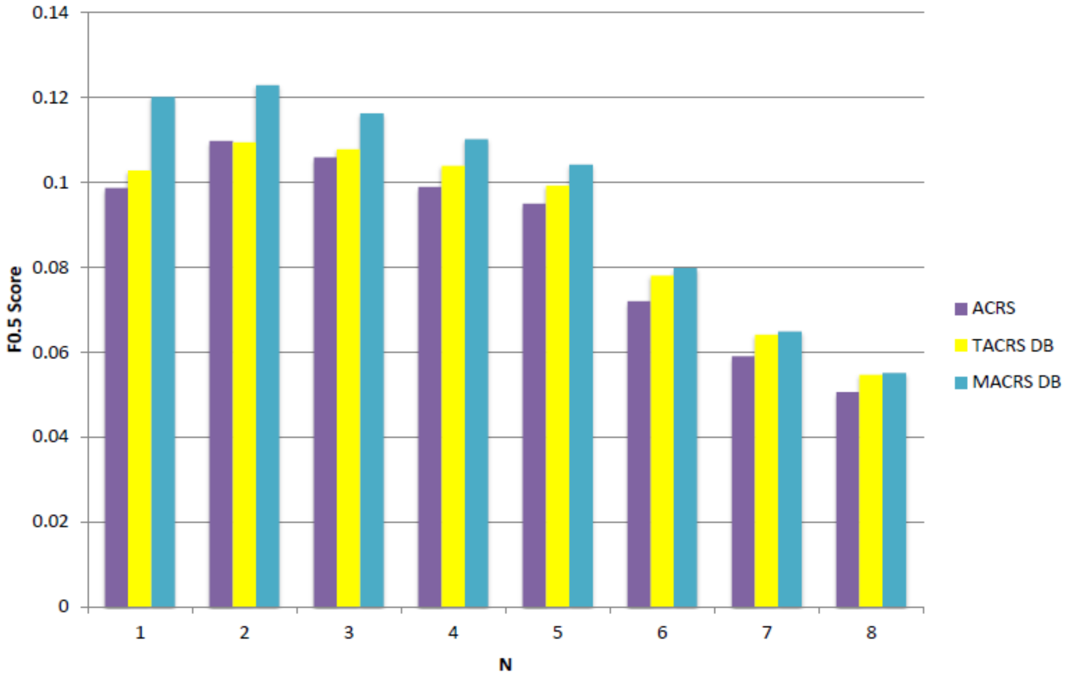

Figure 17. F0.5 score values for *TACRS*$_{DB}$, and *MACRS*$_{DB}$ vs. *ACRS*

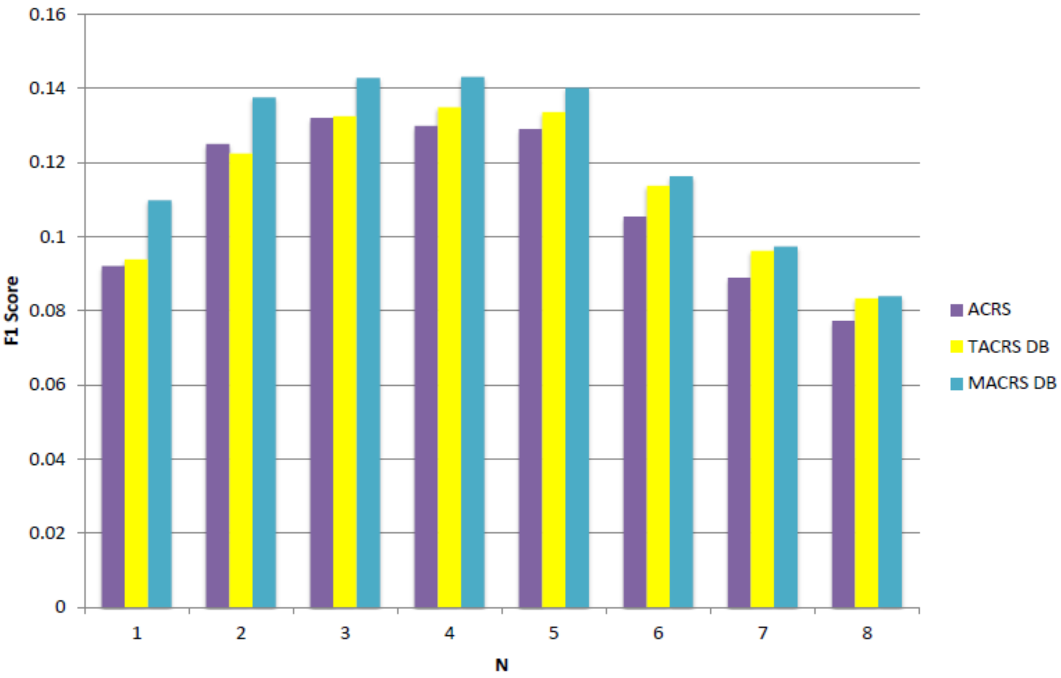

Figure 18. F1 score values for *TACRS$_{DB}$*, and *MACRS$_{DB}$* vs. *ACRS*

5.6.6 Performance of Unified Time-Aware Models

The last two charts in this section as Figure 20 and Figure 21, demonstrate the results of the unified experiment for activity correlation-aware approaches in terms of F0.5 score and F1score respectively. It is noteworthy to point that, the average improvement percentage of UACRS in comparison to ACRS, is around 81% for F0.5 score metric, and 50% for F1 score, which again shows a significant progress for the proposed method as well as the previous approach.

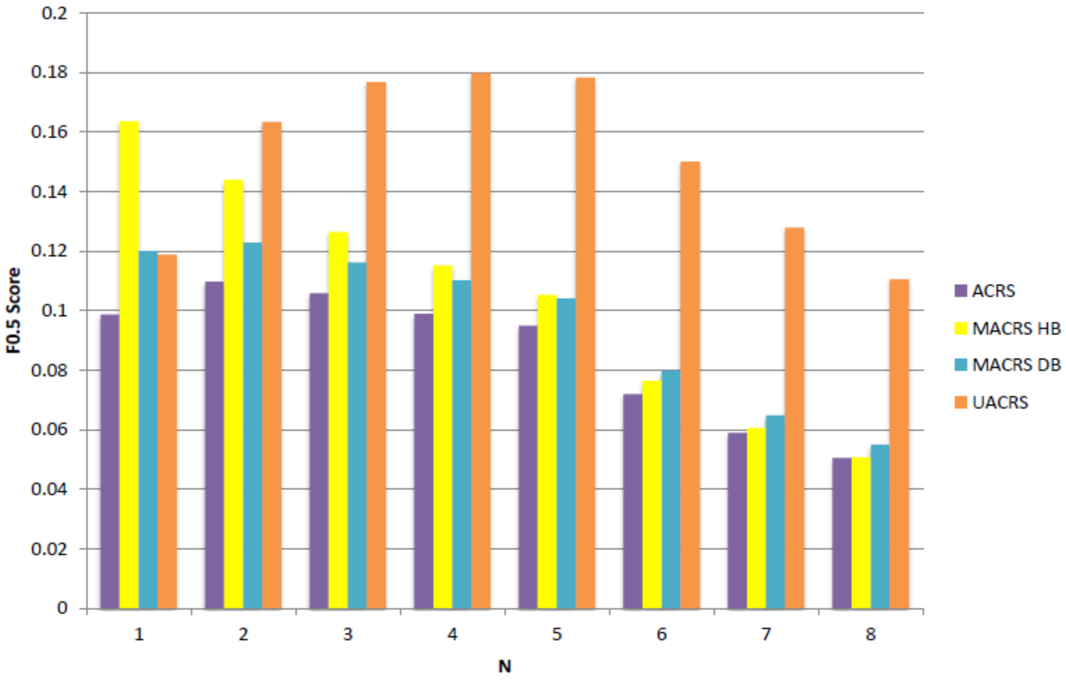

Figure 19. F0.5 score values for *UACRS* vs. *MACRS$_{HB}$* and *MACRS$_{DB}$* and *ACRS*

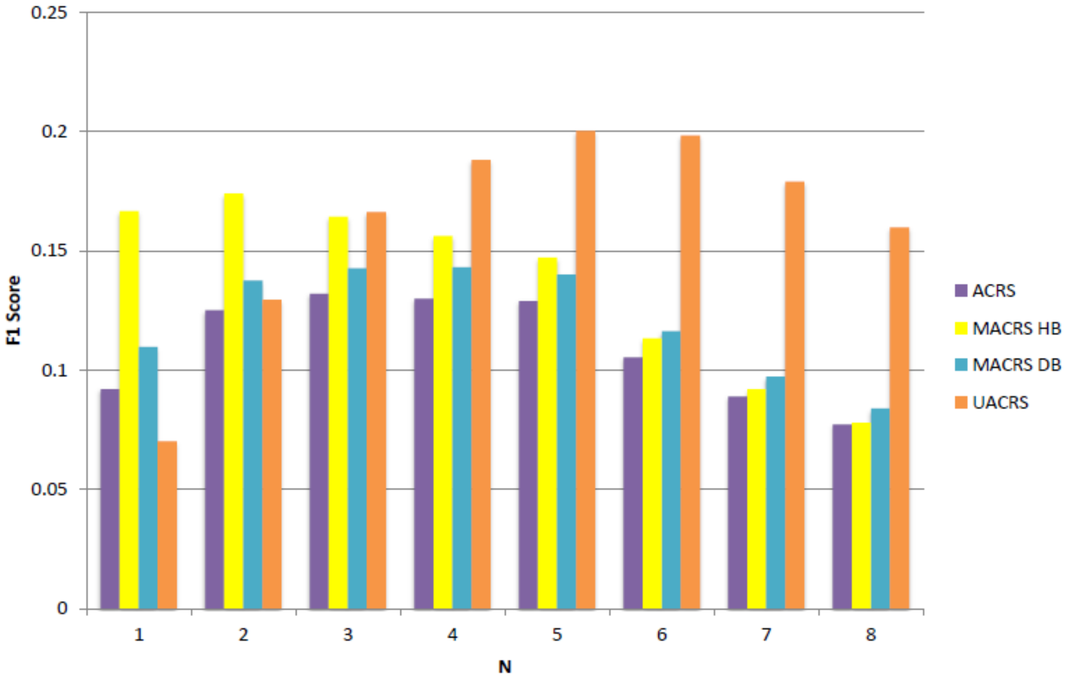

Figure 20. F1 score values for *UACRS* vs. *MACRS$_{HB}$* and *MACRS$_{DB}$* and *ACRS*

5.7 Validation Experiments

In this section the results of the parameter optimization task done for the formulas mentioned in the Chapter 3 and 4 are presented. Figure 22, shows the best value for the parameter β used in Equation 3.29 in SRS method, as the 0.9 when N = 2. Figure 23, also indicates that the best performance can be reached when the parameter β used in Equation 4.7 in TACRS method is 0.7. Since the values produced by all of the formulas are normalized, we considered the range of 0.1 to 0.9 for β.

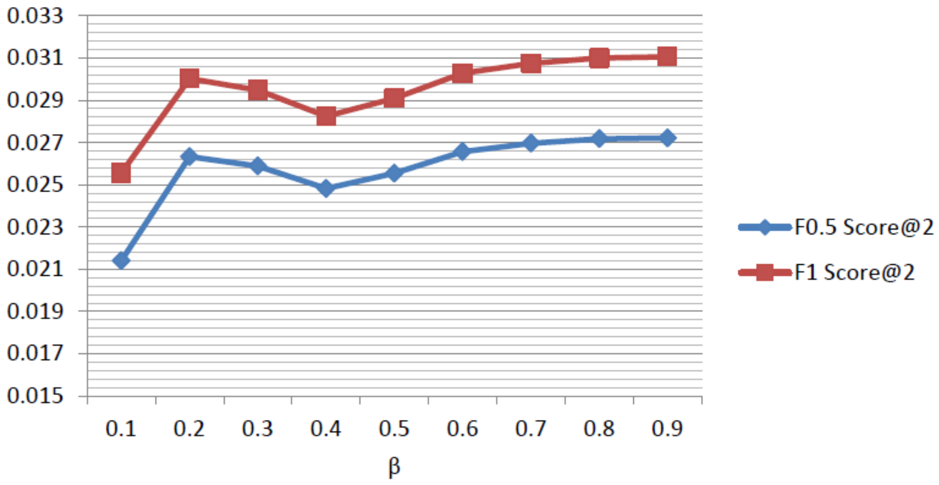

Figure 21. Tuning parameter β used in the *SRS* method

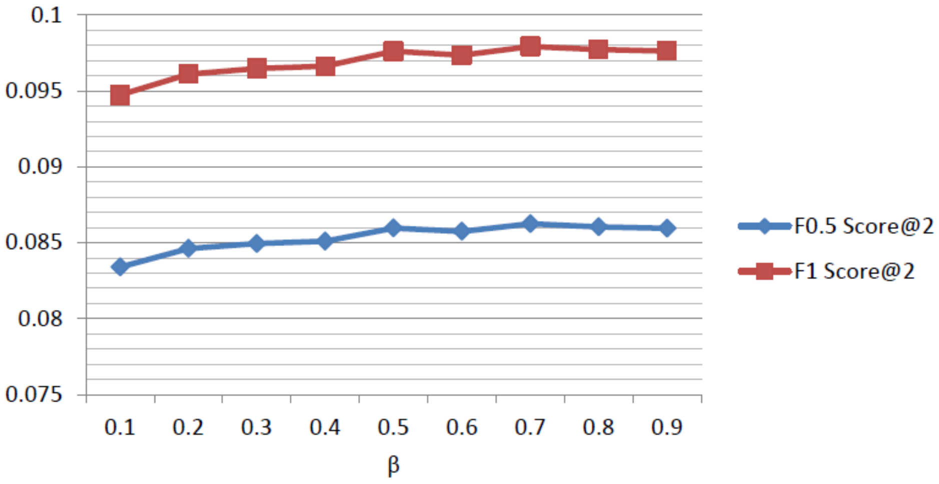

Figure 22. Tuning parameter β used in the *ACRS* and *MACRS* methods

CHAPTER 6

CONCLUSION AND FUTURE WORK

In this thesis, two main activity-based recommendation systems are proposed through improving existing location-based techniques introduced in [11]. Our proposed methods provide user recommendations for the outdoor activities regarding a specific time of the day within a specific day of the week. Since we believe that time has a significant impact on the user behavior for performing an activity, both approaches incorporate temporal factors, with one of them exploiting activity correlation as an additional feature.

We perform our step by step study towards the development for achieving a unified framework on the location-based data set Gowalla, combined with Foursquare in order to include the required activity dimension. As mentioned in the previous chapter, our unified time-aware approach outperforms the basic CF method 75%, and 50% evaluated by the F0.5 score metric and F1 score metric respectively. Similarly, the unified correlation-based approach improves the basic CF method 81%, and 50% with the F0.5 score metric and F1 score metric respectively.

In addition, we introduce two similarity measures from the literature and test their effect on the location-based data to compare with the basic user-based CF recommendation system which uses cosine similarity algorithm for finding similar users. As another minor work we also propose a hybrid model utilizing cascade method for combining two time-aware and space-aware models which are introduced in [11].

Despite the fact that we try to develop a novel multi-feature activity recommendation the techniques in this field still can be improved with several approaches. We offer some of them that can extend the ability and performance efficiency of the proposed system:

- Considering more features such as social links or friendships, exploiting the geographical information of the location and combining them to the activity-aware system, utilizing expert opinions and other possible features to improve the system’s efficiency as a comprehensive framework.
- Concentrating on other novel similarity measures as the core of the collaborative filtering algorithm is one of the promising ideas for increasing the efficiency of the system. As one of the possible options we are interested to use the sequence similarity algorithms which are popular methods in bioinformatics, in order to find similar users based on their activity sequences in a period of time.
- Defining a system, based on different interpretation of the available ratings is another potential field we intend to work on. In our study, we considered binary values as 0 or 1 for ratings. Various interpretations of the ratings such as the triple of like, dislike and unknown as an example creates a new challenge for the future works.

REFERENCES

[1] Sergey Brin and Lawrence Page, "The anatomy of a large-scale hypertextual Web search engine," *Computer Networks and ISDN Systems*, vol. 30, no. 1-7, pp. 107-117, April 1998.

[2] Greg Linden, Brent Smith, and Jeremy York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76-80, January 2003.

[3] Bradley N. Miller, Istvan Albert, Shyong K. Lam, Jodeph A. Konstan, and John Riedl, "MovieLens unplugged: experiences with an occasionally connected recommender system," in *IUI '03 Proceedings of the 8th international conference on Intelligent user interfaces*, New York, 2003, pp. 263-266.

[4] Will Hill, Larry Stead, Mark Rosenstein, and George Furnas, "Recommending and evaluating choices in a virtual community of use," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, 1995, pp. 194-201.

[5] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl, "GroupLens: an open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, York, NY, 1994, pp. 175-186.

[6] Upendra Shardanand and Pattie Maes, "Social information filtering: algorithms for automating “word of mouth”," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, 1995, pp. 210-217.

[7] Vincent W. Zheng, Bin Cao, Yu Zheng, Xing Xie, and Qiang Yang, "Collaborative Filtering Meets Mobile Recommendation: A User-centered Approach," in *AAAI'10*, 2010, pp. 236-241.

[8] Yu Zheng, Xing Xie, and Qiang Yang, "Collaborative location and activity recommendations with GPS history data," in *Proceedings of the 19th*

international conference on World wide web, New York, 2010, pp. 1029-1038.

[9] Mao Ye, Peifeng Yin, Wang-Chien Lee, and Dik-Lun Lee, "Exploiting geographical influence for collaborative point-of-interest recommendation," in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 2011, pp. 325-334.

[10] Jitao Sang, Tao Mei, Jian-Tao Sun, Changsheng Xu, and Shipeng Li, "Probabilistic sequential POIs recommendation via check-in data," in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, 2012, pp. 402-405.

[11] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann, "Time-aware point-of-interest recommendation," in *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, 2013, pp. 363-372.

[12] Paul Resnick and Hal R. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56-58, March 1997.

[13] Robin Burke, "The adaptive web," in *Hybrid web recommender systems*. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 377-408.

[14] Francesco Ricci, Lior Rokach, and Bracha Shapira, "Introduction to Recommender Systems Handbook," in *Recommender Systems Handbook*.: Springer US, 2011, pp. 1-35.

[15] Robin Burke, "Hybrid Recommender Systems: Survey and Experiments," *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331-370, Nov. 2002.

[16] Yi Ding and Xue Li, "Time weight collaborative filtering," in *Proceedings of the 14th ACM international conference on Information and knowledge management*, 2005, pp. 485-492.

[17] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich, *Recommender Systems: An Introduction*.: Cambridge University Press, 2011.

[18] Hao Wang, Manolis Terrovitis, and Nikos Mamoulis, "Location recommendation in location-based social networks using user check-in data," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2013, pp. 374-383.

[19] Sanghack Lee, Jihoon Yang, and Sung-Yong Park, "Discovery of Hidden Similarity on Collaborative Filtering to Overcome Sparsity Problem," in *7th International Conference*, Padova, 2004, pp. 396-402.

[20] Daniar Asanov, "Algorithms and Methods in Recommender Systems," Berlin Institute of Technology, Berlin, 2011.

[21] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th international conference on World Wide Web*, 2001, pp. 285-295.

[22] Michael J. Pazzani and Daniel Billsus, "Content-based recommendation systems," in *The adaptive web*. Berlin: Springer Berlin Heidelberg, 2007.

[23] Andrea Andrenucci , "Collaborative computing," Department of Computer and Systems Sciences - a part of the IT University in Kista, 2006.

[24] Robin Burke, "Knowledge-based recommender systems," *Encyclopedia of Library and Information Systems*, vol. 69, 2000.

[25] Susan Gauch, Mirco Speretta, Aravind Chandramouli, and Alessandro Micarelli, "User Profiles for Personalized Information Access," in *The Adaptive Web*.: Springer Berlin Heidelberg, 2007, pp. 54-89.

[26] Vincent Wenchen Zheng, Bin Cao, Yu Zheng, Xing Xie, and Qiang Yang, "Collaborative Filtering Meets Mobile Recommendation: A User-centered Approach," in *AAAI 2010*, 2010.

[27] Yu Zheng, "Tutorial on Location-Based Social Networks," in *I nproceedings of International conference WWW 2012*, 2012.

[28] Yu Zheng, "Location-Based Social Networks: Users," in *Computing with Spatial Trajectories*.: Springer New York, 2011, pp. 243-276.

[29] Jie Bao, Yu Zheng, and Mohamed F. Mokbel, "Location-based and preference-aware recommendation using sparse geo-social networking data," in *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, 2012, pp. 199-208.

[30] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma, "Mining interesting locations and travel sequences from GPS trajectories," in *Proceedings of the 18th international conference on World wide web*, 2009, pp. 791-800.

[31] Hong Cheng, Jihang Ye, and Zhe Zhu, "What's Your Next Move: User Activity Prediction in Location-based Social Networks," in *Proceedings of the 13th SIAM International Conference on Data Mining*, 2013, pp. 171-179.

[32] Murat Ali Bayir, Murat Demirbas, and Nathan Eagle, "Discovering spatiotemporal mobility profiles of cellphone users," in *World of Wireless, Mobile and Multimedia Networks & Workshops*, 2009, pp. 1-9.

[33] Eunjoon Cho, Seth A Myers, and Jure Leskovec, "Friendship and mobility: user movement in location-based social networks," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 1082-1090.

[34] Anastasios Noulas, Cecilia Mascolo, Salvatore Scellato, and Massimiliano Pontil, "An Empirical Study of Geographic User Activity Patterns in Foursquare," in *Fifth International AAAI Conference on Weblogs and Social Media*, 2011, pp. 570-573.

[35] Wei Fan, "Systematic data selection to mine concept-drifting data streams," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 128-137.

[36] Lin Liao, Dieter Fox, and Henry Kautz, "Location-based activity recognition using relational Markov networks," in *Proceedings of the 19th international*

joint conference on Artificial intelligence, 2005, pp. 773-778.

[37] Vincent Wenchen Zheng, Derek Hao Hu, and Qiang Yang, "Cross-domain activity recognition," in *Proceedings of the 11th international conference on Ubiquitous computing*, 2009, pp. 61-70.

[38] Danny Wyatt, Matthai Philipose, and Tanzeem Choudhury, "Unsupervised activity recognition using automatically mined common sense," in *AAAI'05 Proceedings of the 20th national conference on Artificial intelligence*, 2005, pp. 21-27.

[39] Michael D. Ekstrand, John T. Riedl, and Josep A. Konstan, "Collaborative Filtering Recommender Systems," *Foundations and Trends in Human-Computer Interaction*, vol. 4, no. 2, pp. 81-173, February 2011.

[40] Zheng Hu, Xuzhen Zhu, Ahmad Mian, Hui Tian, and Haifeng Liu, "A new user similarity model to improve the accuracy of collaborative filtering," *Knowledge-Based Systems*, vol. 56, pp. 156-166, January 2014.

[41] Hyung Jun Ahn, "A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem," *Information Sciences: an International Journal*, vol. 178, no. 1, pp. 37-51, January 2008.

[42] Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu, "Fused Matrix Factorization with Geographical and Social Influence in Location-Based Social Networks," *AAAI*, vol. 12, p. 1, 2012.

[43] Betim Berjani and Thorsten Strufe, "A recommendation system for spots in location-based online social networks," in *Proceedings of the 4th Workshop on Social Network Systems*, 2011, pp. 1-6.

[44] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu, "Exploring temporal effects for location recommendation on location-based social networks," in *Proceedings of the 7th ACM conference on Recommender systems*, 2013, pp. 93-100.

[45] Seyyed Mohammadreza Rahimi and Xin Wang, "Location Recommendation Based on Periodicity of Human Activities and Location Categories," in *17th*

Pacific-Asia Conference, PAKDD, 2013, pp. 377-389.

[46] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 5-53, January 2004.

[47] (2015, August) WIKIPEDIA The Free Encyclopedia. [Online]. https://en.wikipedia.org/wiki/F1_score

APPENDIX A

EXPERIMENTAL RESULTS

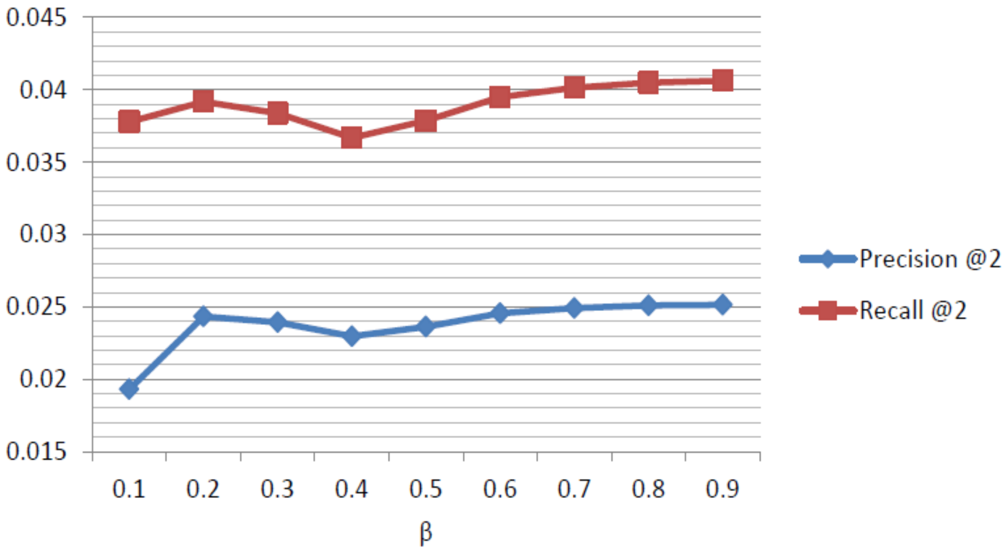

Figure 23. Tuning parameter β used in the SRS method

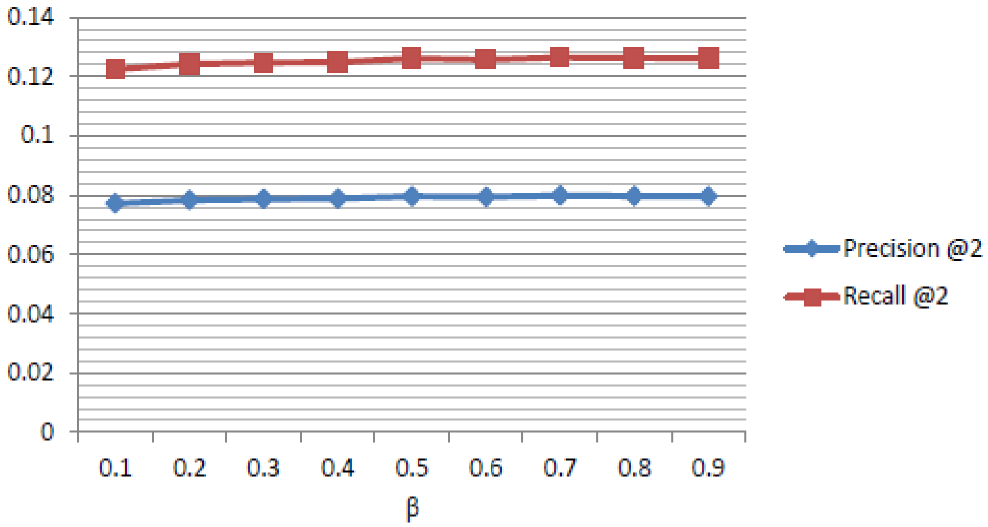

Figure 24. Tuning parameter β used in the TACRS and MACRS methods

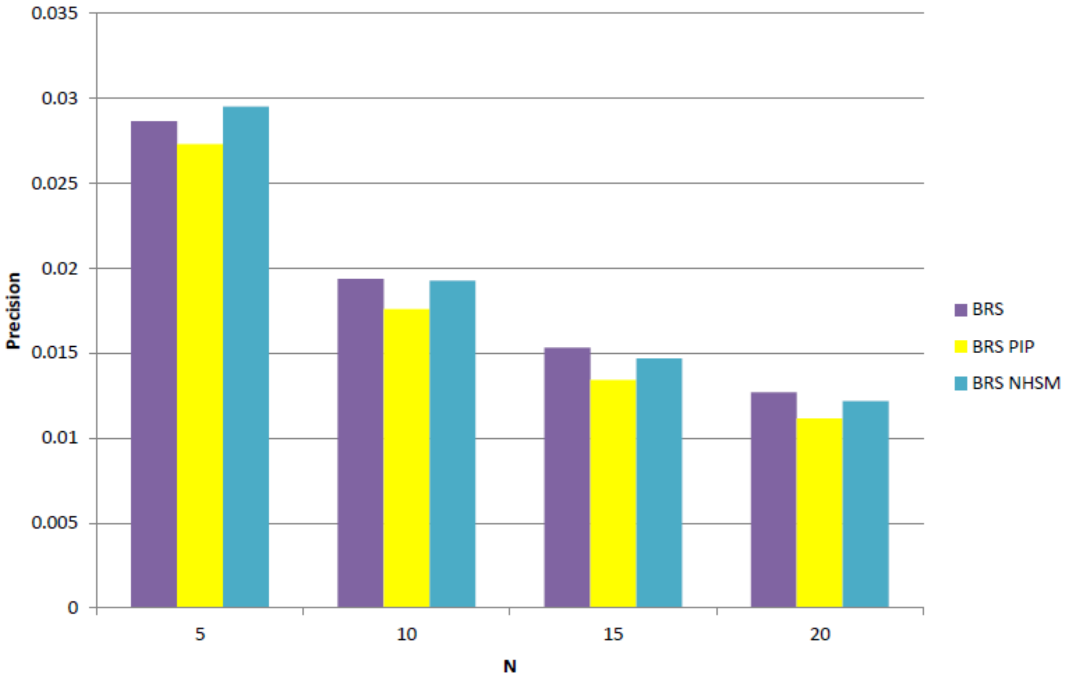

Figure 25. Precision values for the methods BRS_{PIP}, and BRS_{NHSM}, vs. BRS

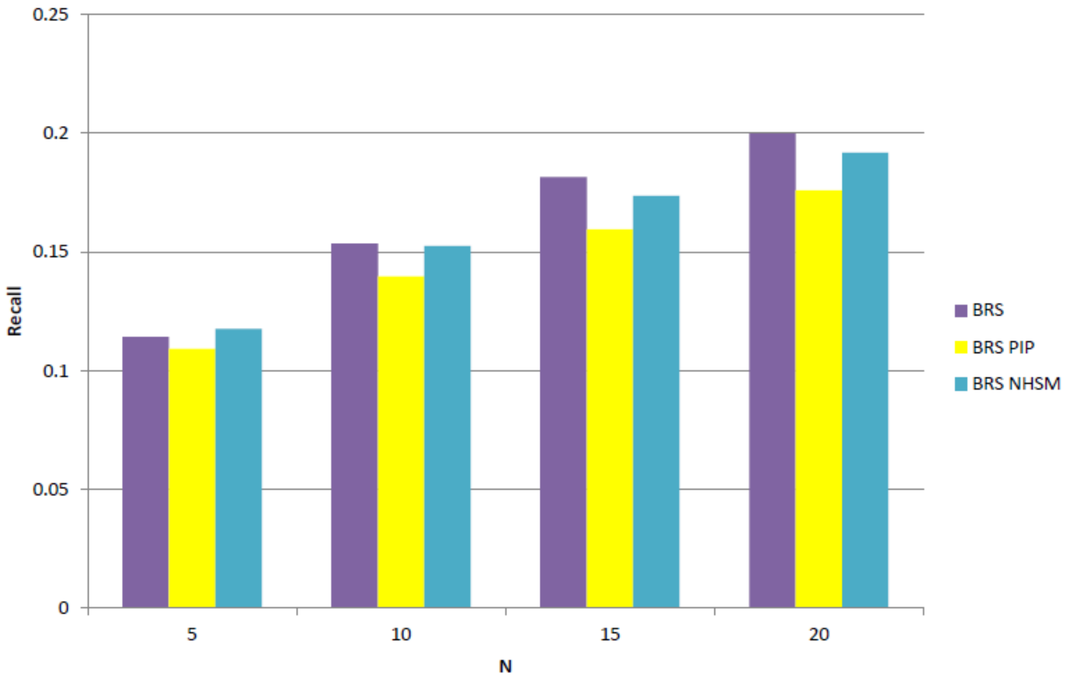

Figure 26. Recall values for the methods BRS_{PIP}, and BRS_{NHSM}, vs. BRS

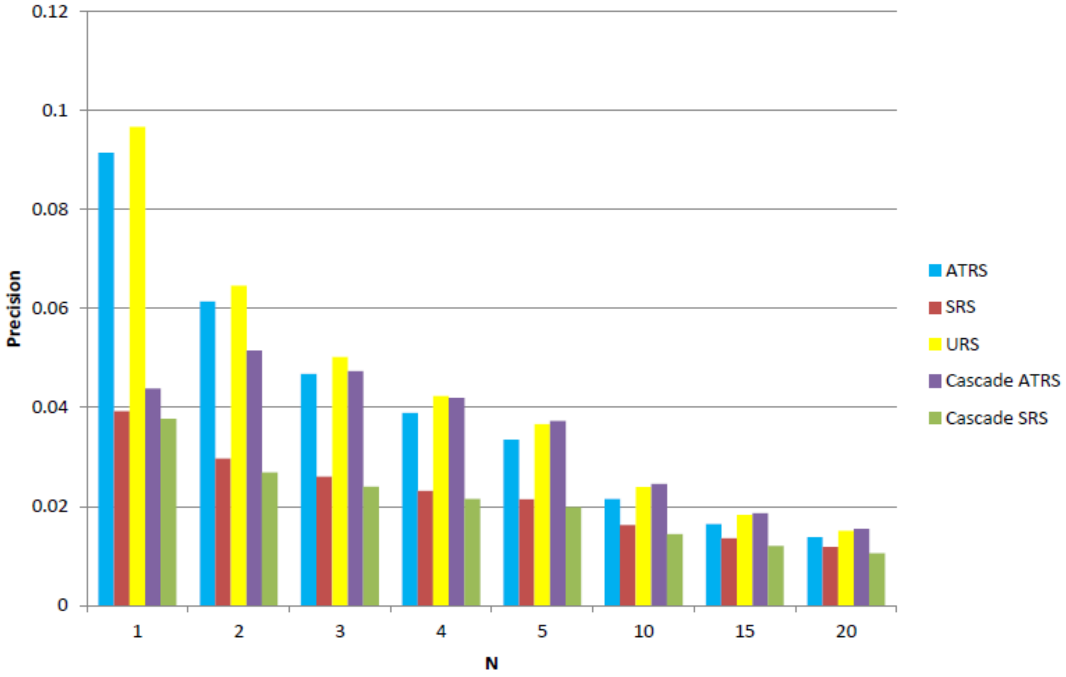

Figure 27. Precision values for *URS, Cascade ATRS, Cascade SRS* vs. *SRS* and *ATRS*

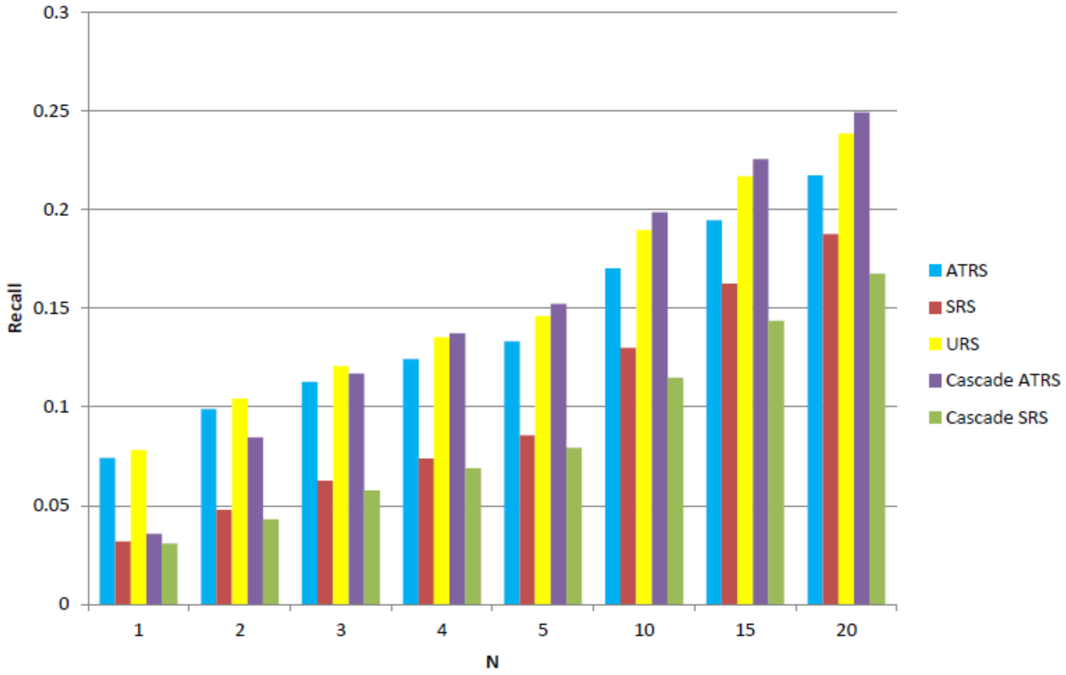

Figure 28. Recall values for *URS, Cascade ATRS, Cascade SRS* vs. *SRS* and *ATRS*

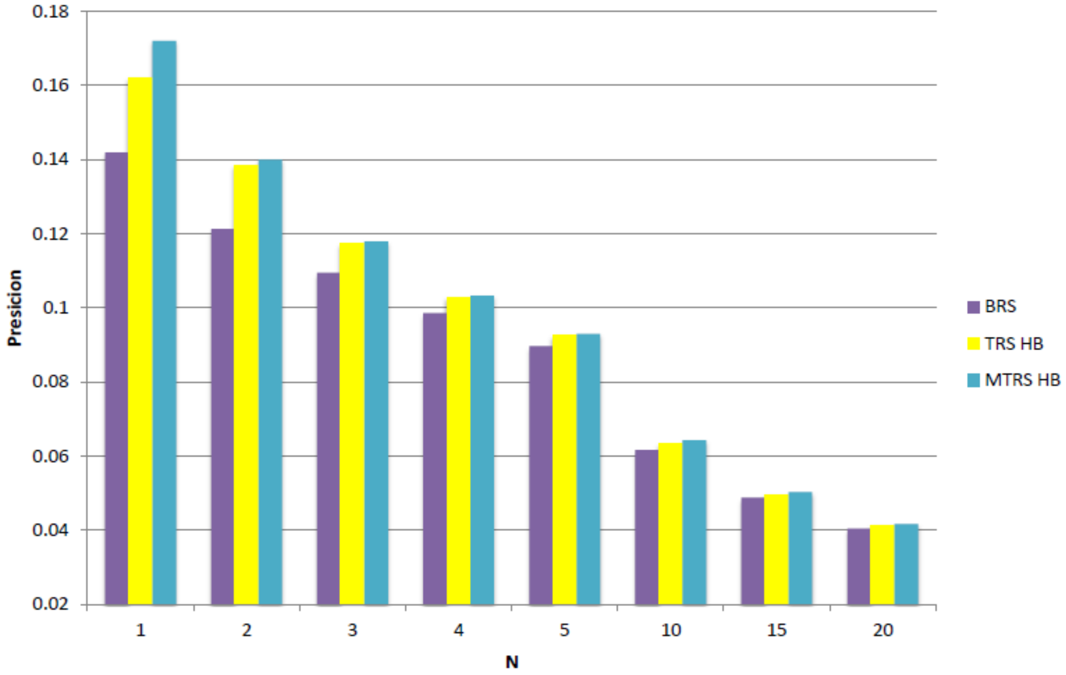

Figure 29. Precision values for *TRS_{HB}*, and *$MTRS_{HB}$* vs. *BRS*

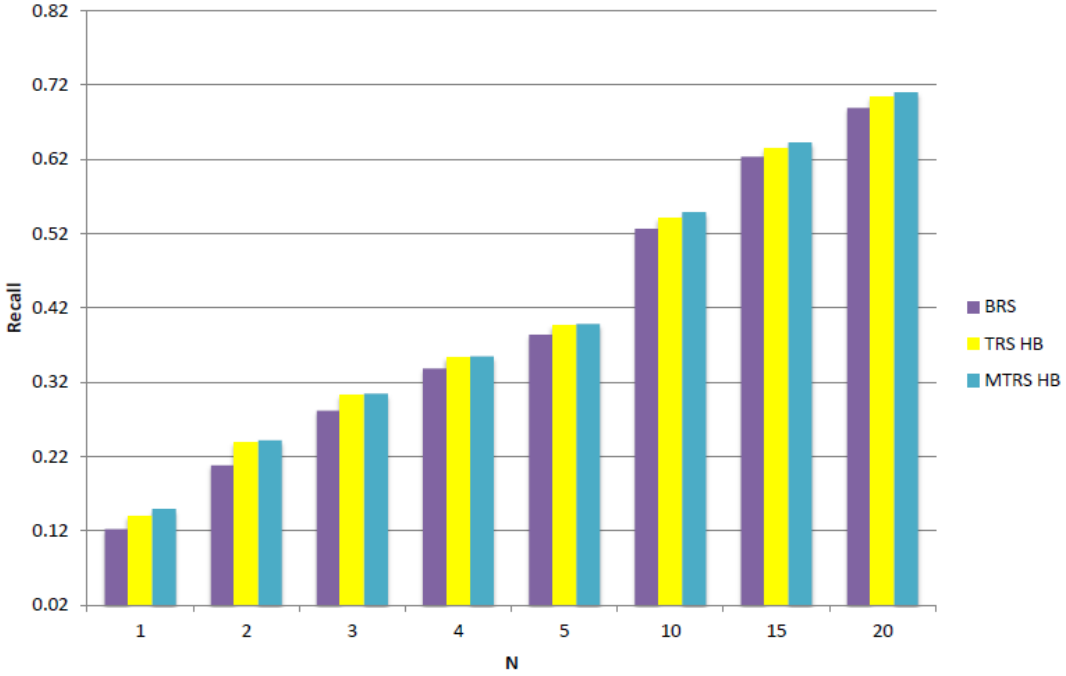

Figure 30. Recall values for *TRS_{HB}*, and *$MTRS_{HB}$* vs. *BRS*

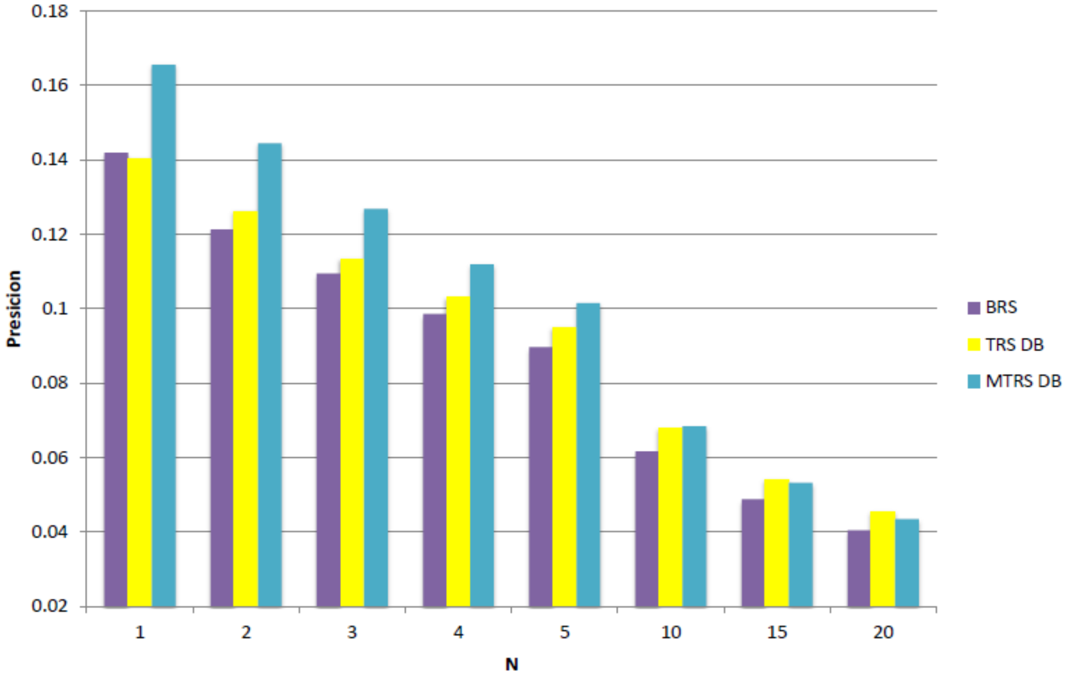

Figure 31. Precision values for *TRS*$_{DB}$, and *MTRS*$_{DB}$ vs. *BRS*

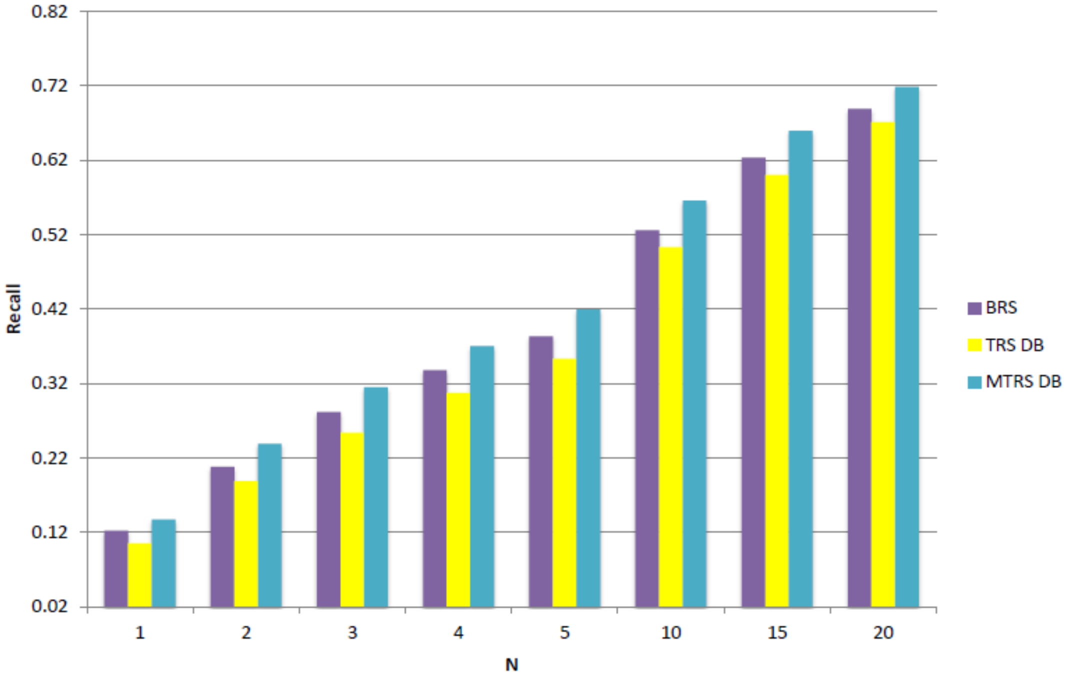

Figure 32. Recall values for *TRS*$_{DB}$, and *MTRS*$_{DB}$ vs. *BRS*

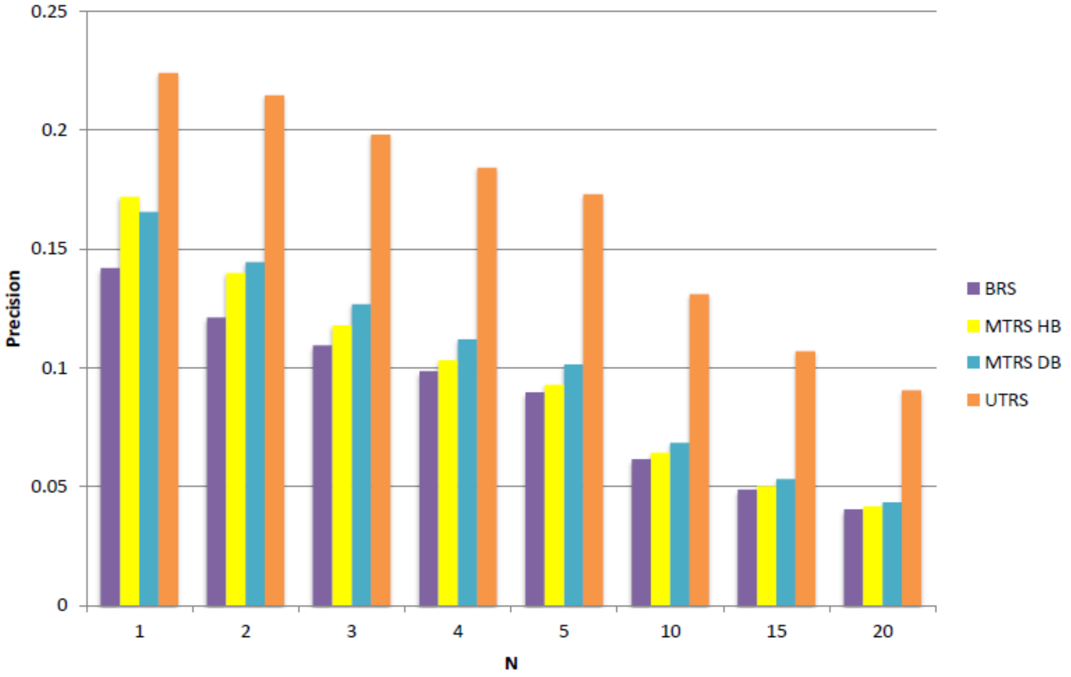

Figure 33. Precision values for *UTRS* vs. *MTRS$_{HB}$* and *MTRS$_{DB}$* and *BRS*

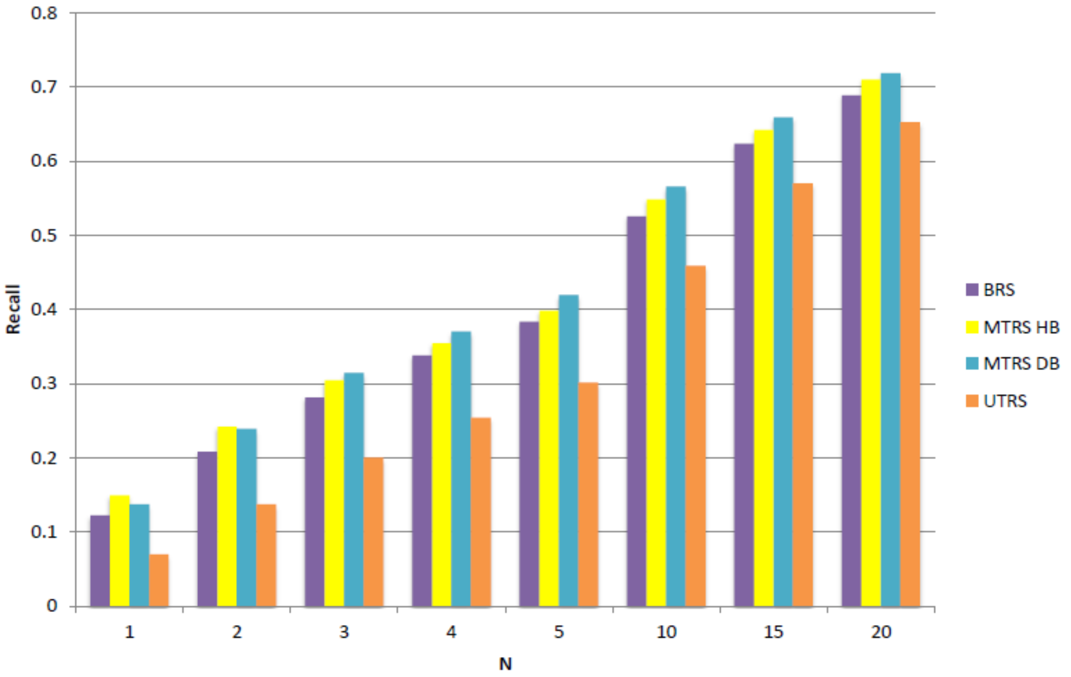

Figure 34. Recall values for *UTRS* vs. *MTRS$_{HB}$* and *MTRS$_{DB}$* and *BRS*

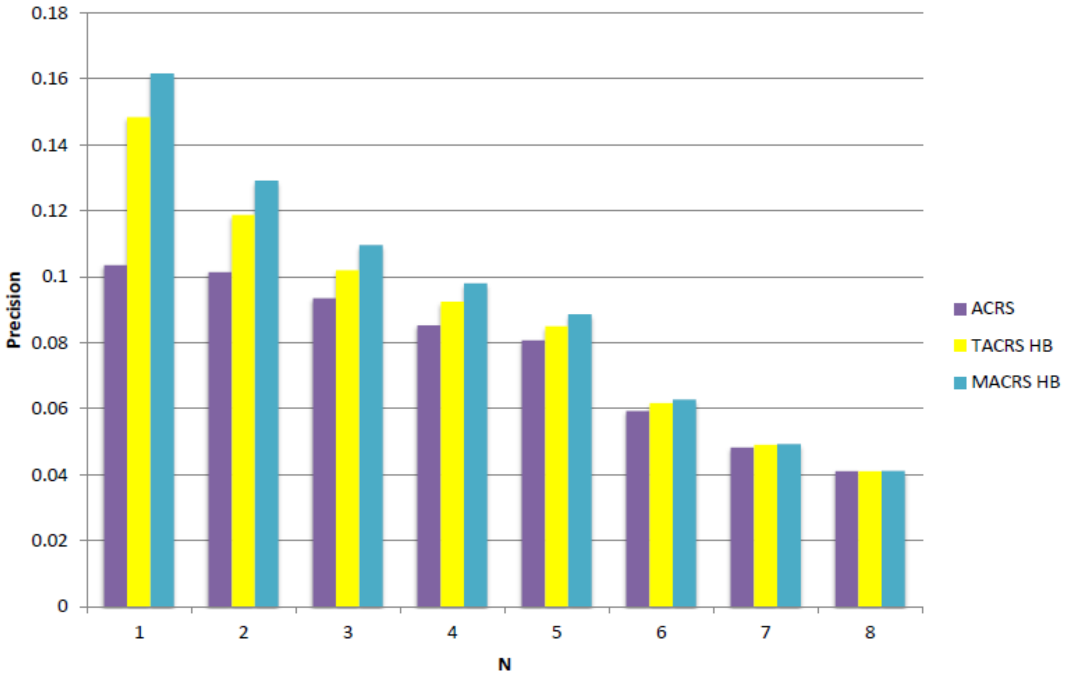

Figure 35. Precision values for *TACRS*$_{HB}$, and *MACRS*$_{HB}$ vs. *ACRS*

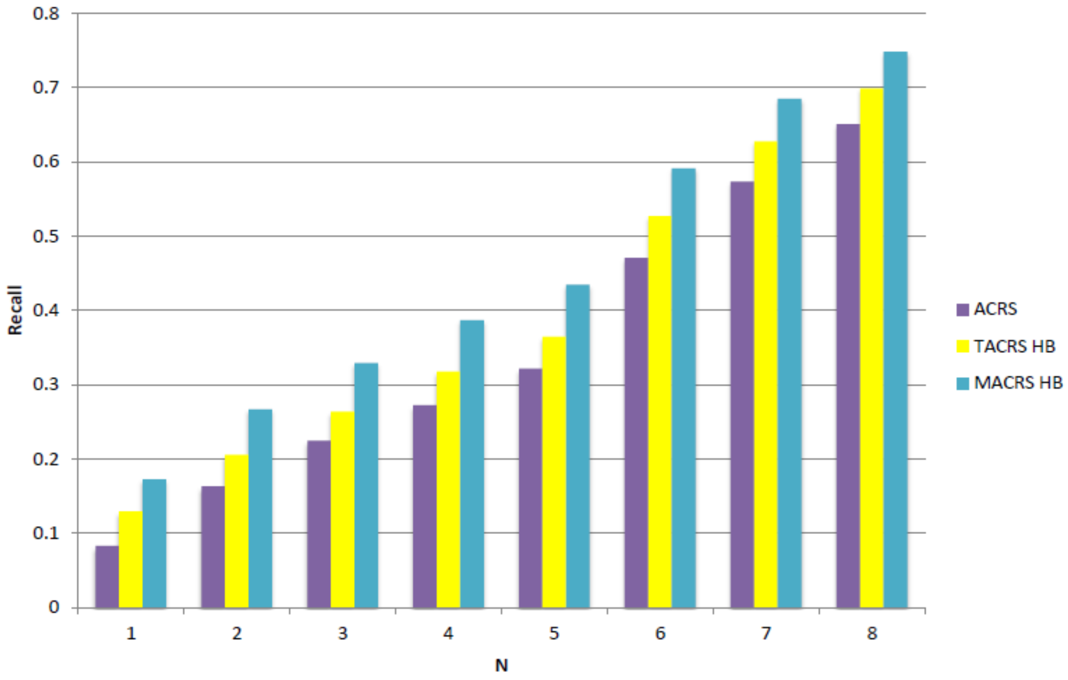

Figure 36. Recall values for *TACRS*$_{HB}$, and *MACRS*$_{HB}$ vs. *ACRS*

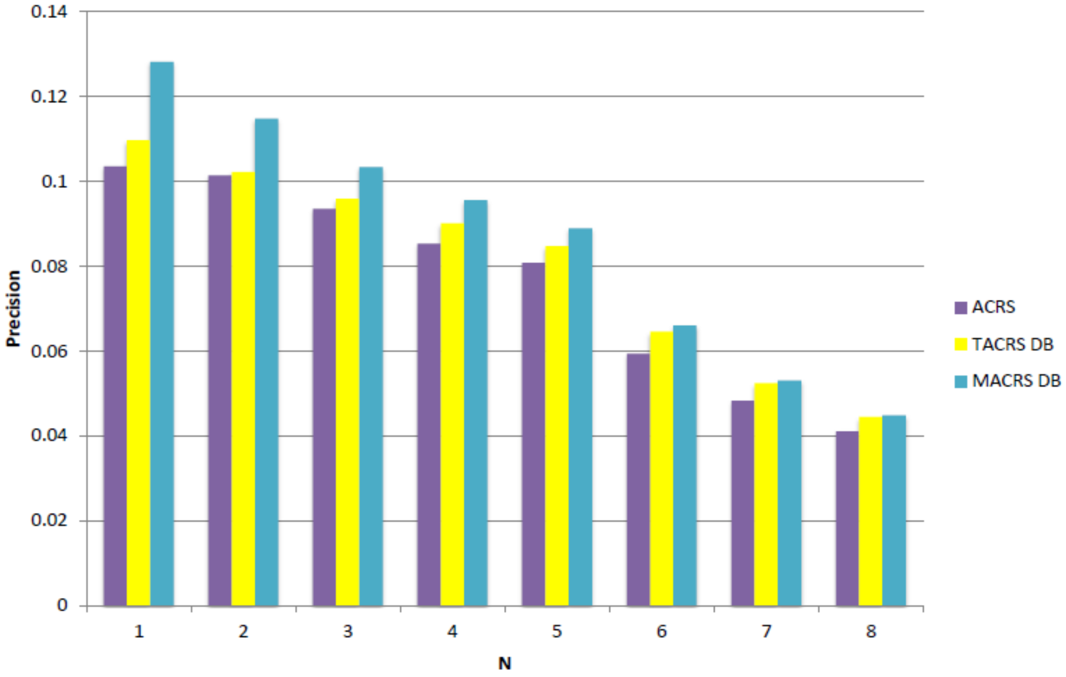

Figure 37. Precision values for *TACRS*$_{DB}$, and *MACRS*$_{DB}$ vs. *ACRS*

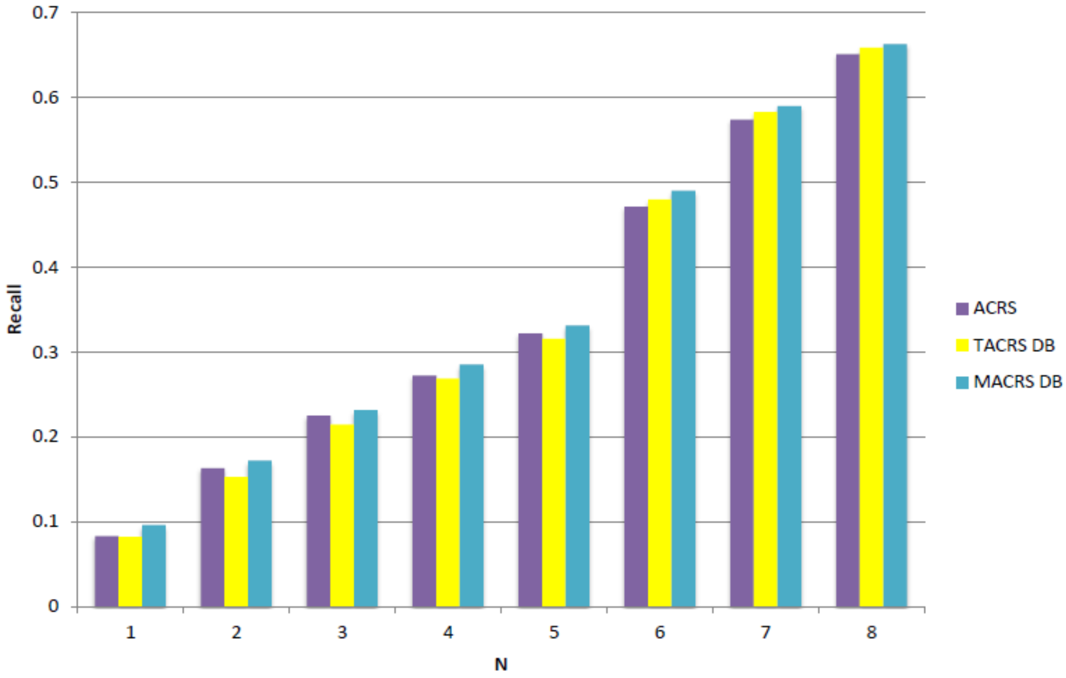

Figure 38. Recall values for *TACRS*$_{DB}$, and *MACRS*$_{DB}$ vs. *ACRS*

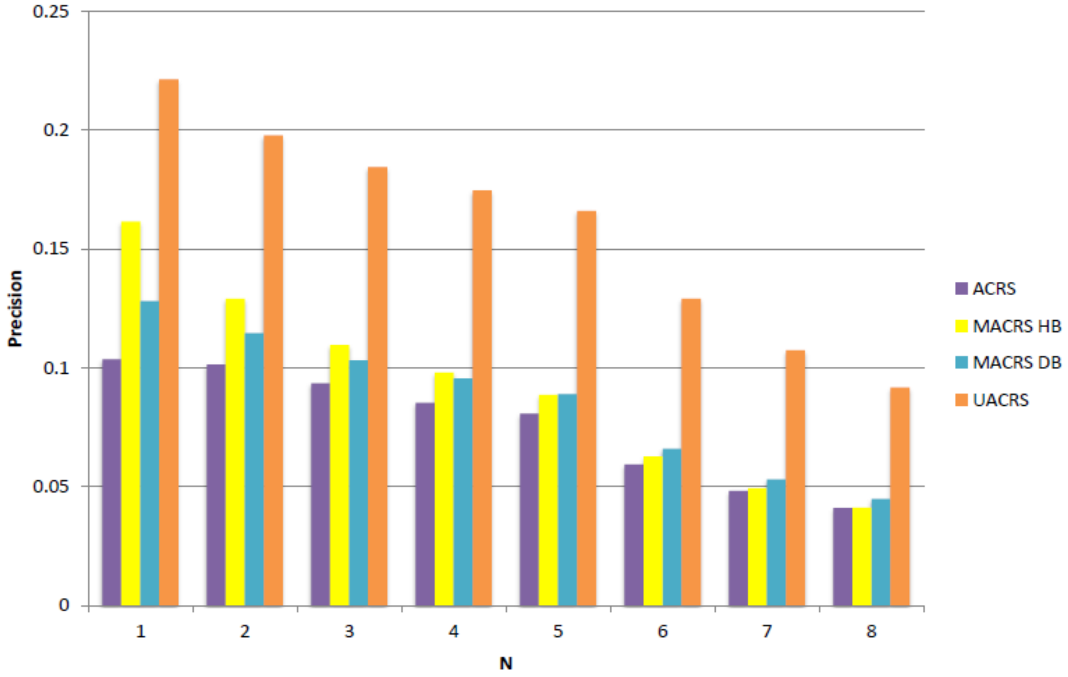

Figure 39. Precision values for *UACRS* vs. $MACRS_{HB}$ and $MACRS_{DB}$ and *ACRS*

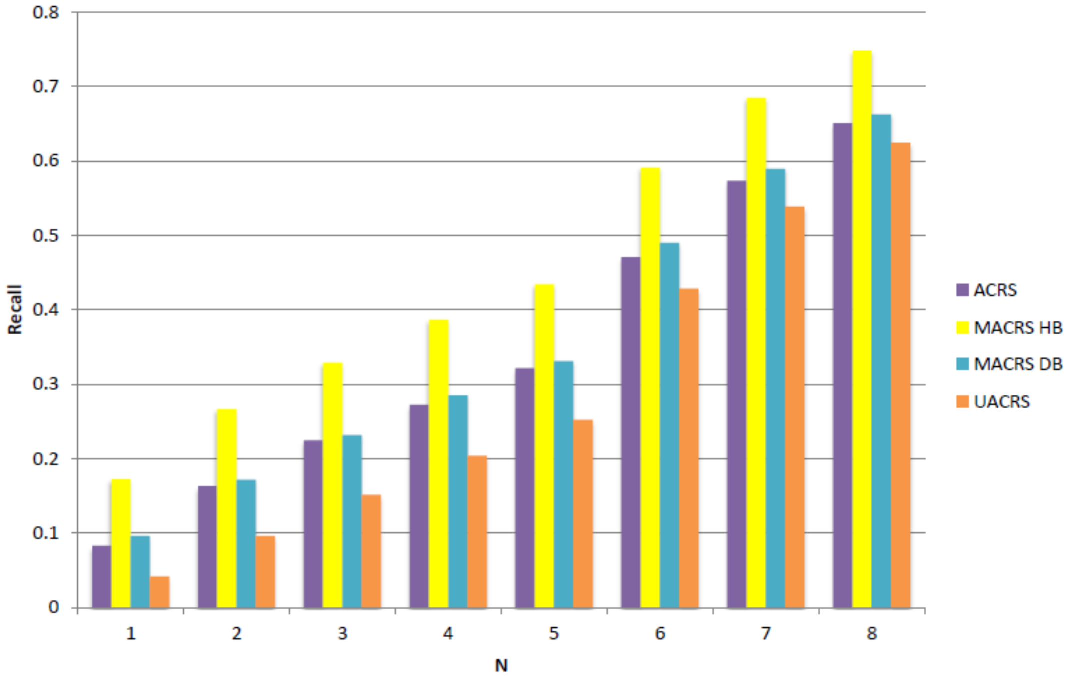

Figure 40. Recall values for *UACRS* vs. $MACRS_{HB}$ and $MACRS_{DB}$ and *ACRS*