KAROL WAGA

Processing, Analysis and Recommendation of Location Data

Publications of the University of Eastern Finland
Dissertations in Forestry and Natural Sciences
Number 195

Academic Dissertation
To be presented by permission of the Faculty of Science and Forestry for public examination in the Auditorium M101 in Metria building at the University of Eastern Finland, Joensuu, on November 9, 2015, at 12 o'clock noon.

School of Computing

Grano Oy
Joensuu, 2015
Editors: Dir. Pertti Pasanen,
Prof. Pekka Kilpeläinen, Prof. Kai Peiponen, Prof. Matti Vornanen
Distribution:
Eastern Finland University Library / Sales of publication
P.O. Box 107, FI-80101 Joensuu, Finland
http://www.uef.fi/kirjasto

ISBN: 978-952-61-1919-9 (printed)
ISBN: 978-952-61-1920-5 (pdf)
ISSNL: 1798-5668
ISSN: 1798-5668
ISSN: 1798-5676 (pdf)

Author's address:

University of Eastern Finland
School of Computing
P.O.Box 111
80101 Joensuu
FINLAND
email: karol.waga@uef.fi

Supervisor:

Professor Pasi Fränti, Ph.D.
University of Eastern Finland
School of Computing
P.O.Box 111
80101 Joensuu
FINLAND
Email: pasi.franti@uef.fi

Reviewers:

Dr. Jukka Teuhola
Department of Information Technology
University of Turku
FINLAND
Email: teuhola@it.utu.fi

Dr. Shonali Krishnaswamy
Data Mining Department
Institute for Infocomm Research (I2R)
SINGAPORE
Email: spkrishna@i2r.a-star.edu.sg

Opponent:

Professor Vasile Manta, Ph.D.
Technical University of Iași
Faculty of Automatic Control and Computer Engineering
Department of Computer Engineering
Blvd. D. Mangeron 53A
700050 Iași
ROMANIA
Email: vmanta@cs.tuiasi.ro

ABSTRACT

This thesis describes the processing, analysis and recommendation of location data. Firstly, the efforts to create an efficient and complete system for handling GPS trajectories are presented. The proposed system allows for the effective storage and visualization of trajectories as well as their analysis including segmentation and detection of movement type. Secondly, a recommendation system is described. It recommends what users should do next in their current location based on geotagged photos and GPS trajectories collected by other users. Thirdly, the methods for calculating user similarity are presented and evaluated. The resulting similarity scores are designed to personalize the recommendation system.

Universal Decimal Classification: 004.6, 004.738.5, 004.775, 004.78, 528.021.6

Library of Congress Subject Headings: Recommender systems (Information filtering); Location-based services; Wireless localization; Global Positioning System; Information visualization; Electronic data processing; Online social networks

Yleinen suomalainen asiasanasto: suosittelujärjestelmät; paikkatiedot; paikannus; satelliittipaikannus; personointi; visualisointi; sosiaalinen media

Keywords: recommender, recommendation system, GPS trajectory, user similarity, personalization, location-based services, travel mode detection

ACKNOWLEDGEMENTS

The work presented in this thesis has been carried out in Speech and Image Processing Unit at the School of Computing, University of Eastern Finland, Joensuu, Finland, during the years 2011–2015.

I would like to thank my supervisor, Professor Pasi Fränti, for support, ideas and guidance over the years. I am also thankful to my colleagues who helped in my research by useful suggestions and discussions.

I am thankful to Dr. Jukka Teuhola and Dr. Shonali Krishnaswamy, the reviewers of the thesis, for the constructive feedback and comments. I would like to thank Professor Vasile Manta for acting as my opponent.

This research has been supported by Cross Border University (CBU), Nokia Foundation and MOPSI project.

I would like to thank my fiancée, Katalin Kiss, for her encouragement to work on the thesis and her continous support in life. Without her I would not be the same person and this thesis would not be possible. I am grateful to my parents who always encouraged me to follow my own path and who gave me moral support during the work on this thesis. I appreciate all the time I spent with my friends, especially Hilla Pihajoki and Radu Mariescu-Istodor, who made my life in Joensuu a great adventure that helped me to find energy to work on the research.

Joensuu, 10.10.2015
Karol Waga

LIST OF ABBREVIATIONS

AGPS	Assisted Global Positioning System
API	Application Programming Interface
EMD	Earth Mover Distance
GPS	Global Positioning System
GSM	Global System for Mobile Communications
kNN	k Nearest Neighbors

LIST OF SYMBOLS

$\sigma(V_j)$	speed variance of the segment *j*
I	set of items *i*
m_i	state of *ith* segment
N	normalized score of item *s* from set *S*
p_i	*ith* point from GPS trajectory
S	set of scores s of items *i*
t_i	timestamp of the *ith* point from GPS trajectory
V_j	set of speeds of segments *j*
x_i	latitude of the *ith* point from GPS trajectory
X_i	feature vector of the *ith* segment
y_i	longitude of the *ith* point from GPS trajectory

LIST OF ORIGINAL PUBLICATIONS

This thesis is the review of author’s work in the field of recommendation systems and GPS trajectory analysis and the following selection of the author’s publications.

I K. Waga, A. Tabarcea, R. Mariescu-Istodor and P. Fränti, "Real Time access to multiple GPS tracks", *9th Int. Conf. on Web Information Systems & Technologies (WEBIST'13),* pp. 293–299, Aachen, Germany, May 2013.

II K. Waga, A. Tabarcea, M. Chen and P. Fränti, "Detecting movement type by route segmentation and classification", *8th IEEE Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom'12),* pp. 508–513, Pittsburgh, USA, October 2012.

III K. Waga, A. Tabarcea and P. Fränti, "Recommendation of points of interest from user generated data collection", *8th IEEE Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom'12),* pp. 550–555, Pittsburgh, USA, October 2012.

IV P. Fränti, K. Waga and C. Khurana, “Can social network be used for location-aware recommendation?”, *11th Int. Conf. on Web Information Systems & Technologies (WEBIST'15),* pp. 558–565, Lisbon, Portugal, May 2015.

V K. Waga and P. Fränti, "Similarity of mobile users based on sparse location history", manuscript (submitted)

Throughout the overview the above publications are referred to as [I]–[V]. The publications have been included in the thesis with permission of their copyright holders.

AUTHOR'S CONTRIBUTION

In paper [I] the author designed the GPS trajectories visualization system following suggestions of other authors and implemented it as a part of the Mopsi project. In paper [II] the author worked together with the two other authors on enhancement of the segmentation and classification algorithms proposed originally by Dr. Chen. The author provided first implementation of the user interface for the algorithms visualizing the results on map in Mopsi. In paper [III], the idea of the recommendation system originated from the author of this thesis. The author implemented it as part of the Mopsi system. The author proposed the scoring function following advices of Prof. Fränti. Prof. Fränti wrote paper [IV] showing two complementary ideas on measuring similarity between users using social network data and sparse location data. The author proposed and was responsible for the method utilizing location. In paper [V], the author proposed to measure similarity of users based on sparse location data and Prof. Fränti helped to analyze the results of different similarity measurement methods. The user similarity problem was transformed into analysis of histogram similarity problem. The author implemented and tested all the methods as well as was responsible for majority of writing process, except for paper [IV].

Contents

1 Introduction

1.1 MOTIVATION

Personalized services play important role in everyday life. Personalization is included in websites: search results, suggested friends, recommended items for purchase are only few examples. A common approach to personalization is to utilize explicit data about the user, such as profile preferences or ratings given. However, implicit data, such as purchase history or pages viewed, is also used.

Increasing the availability of mobile devices resulted in the emergence of a new market for personalized applications. Location became an important factor for personalization because positioning techniques such as GPS are commonly installed in mobile devices. The majority of mobile devices also have a connection to the Internet.

In this dissertation, we present a contribution to the field of location-based recommendation applications and services. The main contribution is a recommendation system that suggests what users should do in their current location. The recommendation system is a complex system that can be further enhanced by applying tools and algorithms developed during this study. GPS trajectory segmentation and movement type classification can be used to recommend destinations based on the current transportation mode used by the user. For better access to the GPS trajectories, a system was developed to access multiple GPS tracks in real time. Work carried out on the similarity of users can further personalize the recommendation system and enhance user experience.

1.2 LOCATION-BASED APPLICATIONS

The increasing availability of smartphones with Internet access and other technologies for users to interact with web-based services and location-based applications has gained in popularity with mobile users [55]. There have been many location-based applications for a long time. Popular examples are Google Maps as well as other similar maps and localized search services. The location concept is present in many popular social applications, such as Foursquare and Facebook. Location-based applications offer users personalized service and provide context-aware information [71]. One popular example is a request for a weather forecast, which can be automatically adjusted to the user's current location. In social networking, personalization might come in the form of notifications when a user's friends appear in nearby locations [55]. There are also location-based applications that provide multiple types of contextualized information. For instance, Foursquare shows the movement of users and his or her social network and, at the same time, shows available nearby services for all users. From a developer's point of view, creating a system that uses multiple location-based applications is not trivial because the developed application easily becomes tightly connected to particular platforms and APIs [55].

1.3 RESEARCH CHALLENGES

There is variety of location-based applications. In this study, we focus on location-based recommendation systems as the author developed a context-aware recommendation system using a collection that had been generated by users.

The main challenge is to select relevant items from the collection of location data such as services, geotagged photos and GPS trajectories with minimal or no user input at all. Nevertheless, the research is not limited to recommendation systems. As we discovered during the studies, to build an efficient recommendation system that suggested to users where they should go next by selecting trajectories toward attractive destinations, it was necessary to find a way to efficiently store and analyze the trajectories in order to get more complex statistics such as method of travel. That resulted in research on real-time storage and a visualization of the trajectories as well as filtering, segmentation and movement type classification.

Another challenge was personalization of the system: user profiles needed to be designed. In addition, the author of this thesis created user-similarity measures in order to find out more about similar users based merely on their activity within the system. The result of the study is an existing location-based system that processes, analyzes and recommends location data.

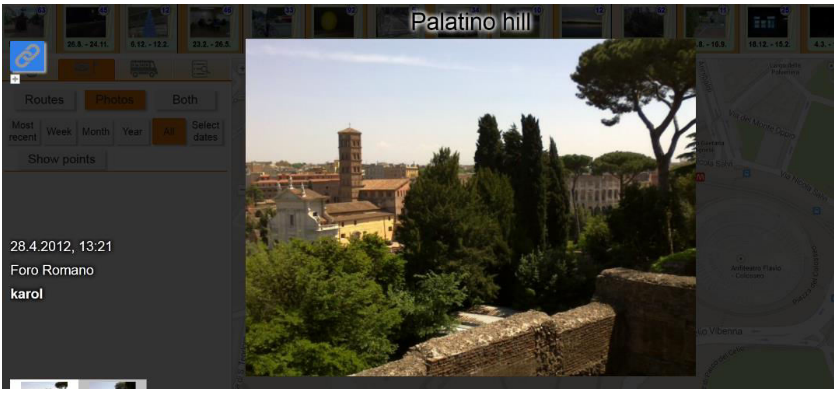
2 Location Data

2.1 GEOTAGGED PHOTOS AND GPS TRAJECTORIES

The wide availability of mobile devices equipped with a positioning function (for example GPS and AGPS) and Internet connectivity has enabled location-based services to become ubiquitous [24]. Such services operate by location data. Location data includes, but is not limited to, geotagged photos and trajectories. Other examples of location data are points of interest and services as restaurants, shopping centers and health centers. The thesis focuses on geotagged photos and GPS trajectories (referred to later as trajectories).

Geotagging is the process of adding location data to various media such as photos and videos. The most common form of geotagging is latitude and longitude, but the information in a geotag can also be enhanced by altitude, bearing and accuracy, among others.

All the photos used in this thesis are geotagged by mobile applications using latitude, longitude and, where available, street addresses. In addition to geotagging information, the photos have timestamps. An example of such a geotagged picture is shown in Fig. 1.



28.4.2012, 13:21
Foro Romano
karol

Figure 1. *Example of a geotagged picture from a user's collection in the Mopsi system.*

As with geotagged photos, GPS trajectories similarly show the location of users. The trajectories consist of a set of points that are ordered in a sequence. The ordering is based on timestamp of each point. In this thesis, we consider GPS trajectories formed of points that contain a user's location at certain time. More precisely, a point p_i in the trajectory is represented as triple $p_i=(x_i, y_i, t_i)$, where x_i is the latitude, y_i is the longitude and t_i is the timestamp of the point [57]. The GPS trajectory is defined as a sequence of such points p_i, $i=0,1,2,\ldots,n$ and for all $0\leq i\leq n$, $t_{i+1}>t_i$ [86]. The trajectory ends when the difference between timestamps of two consecutive points is larger than the threshold Δt [97]. Examples of different visualized GPS trajectories are shown in Fig. 2.

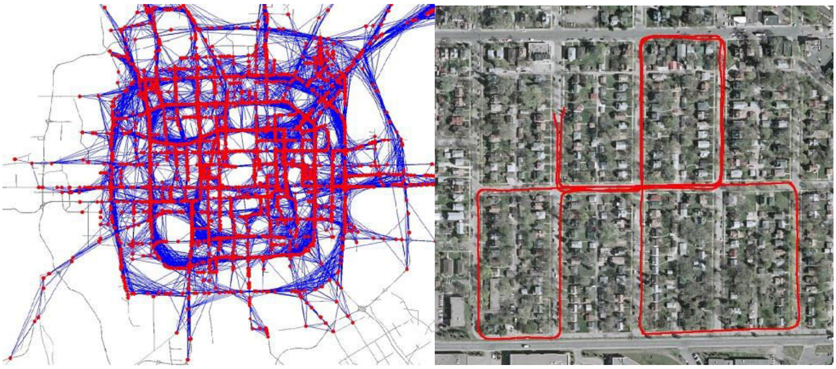


Figure 2. *Different approaches to GPS trajectory visualization on map. Microsoft Research – left, Multiple Autonomous Robotic Systems Laboratory – right.*

Both geotagged photos and their GPS trajectories form the location history of users. Following the definition in [97], we can define location history in this thesis as a set of locations visited by the user over a certain period of time. In practice, we consider locations where photos have been taken, and the start- and end points of the trajectories. In some cases, entire trajectories are used, for example, in the movement type analysis.

2.2 MOPSI SYSTEM

The work described in this dissertation has been implemented and tested within Mopsi system, which has the research of location-based applications as its focus. The Mopsi system was developed by the Speech and Image Processing Group from the School of Computing at the University of Eastern Finland and offers various functionalities. The system is used as a testing workbench for new research solutions in area of location-based applications.

The Mopsi website and mobile applications allow for the collection of location-based data such as photos and GPS trajectories as well as provide tools to browse and analyze them. In addition, Mopsi offers integration with the popular social network, Facebook. Mopsi has a website and mobile applications for major platforms: Android, WindowsPhone, iOS and Symbian.

Photos are uploaded to the Mopsi system using the mobile application, where each user can create his or her photo collections. Sharing the location of the photos is very easy as the photos are automatically geotagged. The uploaded photos are presented to users on a map.

Besides geographical location, each photo has a timestamp. Descriptions can also be provided. Photos can be shared with other users of the system who can browse photos conveniently using time query filtering. The photos matching search criteria are shown on a map and timeline as demonstrated in Fig. 3. The data collected by mobile clients are then presented to users on the website. The applications collect geotagged photos and GPS trajectories. Besides these the applications, users may communicate by chat and O-Mopsi location-based game outlets.

The Mopsi environment has over 2,400 registered users; the database contains over 35,000 photos and over 10 million GPS points that form about 10,000 trajectories. In addition to the user-generated collection of photos and trajectories, the database has information about over 600 points of interests, i.e. services, such as shops, restaurants and tourist attractions, among others. The points of interests are mainly located in Joensuu, Finland, from where Mopsi originates. The services are created and verified by trusted users who collect a significant amount of good quality photos and trajectories to the system.

2.3 RESEARCH AREAS IN MOPSI

The Mopsi system is a real-time working environment where research ideas are tested. The main research topics in the Mopsi system are the collection and analysis of location-based data, web mining [32][74], storage, display, analysis and the compression of GPS trajectories [81][I], detection of transportation mode [II], recommendation of points of interests and things to do next within the user's current location [80][III], and location-based games [74]. We designed a system for efficient real-time retrieval and display of the trajectories [81][I]. The fast retrieval system is based on the polygonal approximation method [20]. The trajectories are stored in compressed form to save storage space [21]. The trajectories are then analyzed and the transportation mode is detected based on various basic characteristics of the trajectories [81][II]. A low complexity algorithm has also been designed to compute the spatial similarity of the trajectories [57]. The real-time analysis of GPS trajectories coming to server allows for the detection of several types of events such as user meetings or visits at certain locations from the Mopsi database [56]. We have used the Mopsi system as a framework to test our personalized recommendation system [80][III] that recommends places and items from the Mopsi database of photos and trajectories considering identified aspects of relevance [30].

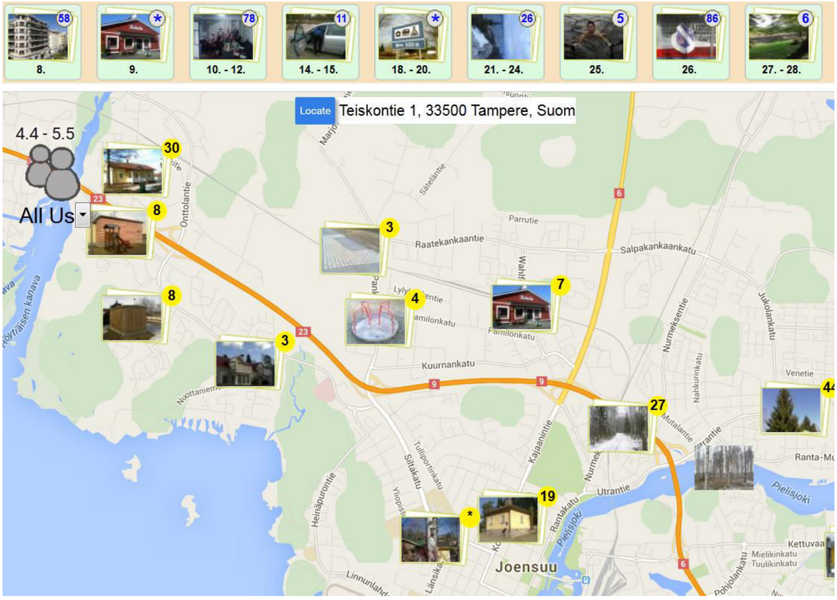


Figure 3. Collection of photos in the Mopsi system collected by all users in Joensuu area within a single month.

As with the photos, GPS points are similarly uploaded to the Mopsi system by using the tracking option in the mobile application. The GPS points are then stored in the database and grouped into trajectories. The trajectory is automatically created based on the timestamp and location of geographical points that have been uploaded to server. The collection of trajectories can be browsed on the map by time as shown in Fig. 4.

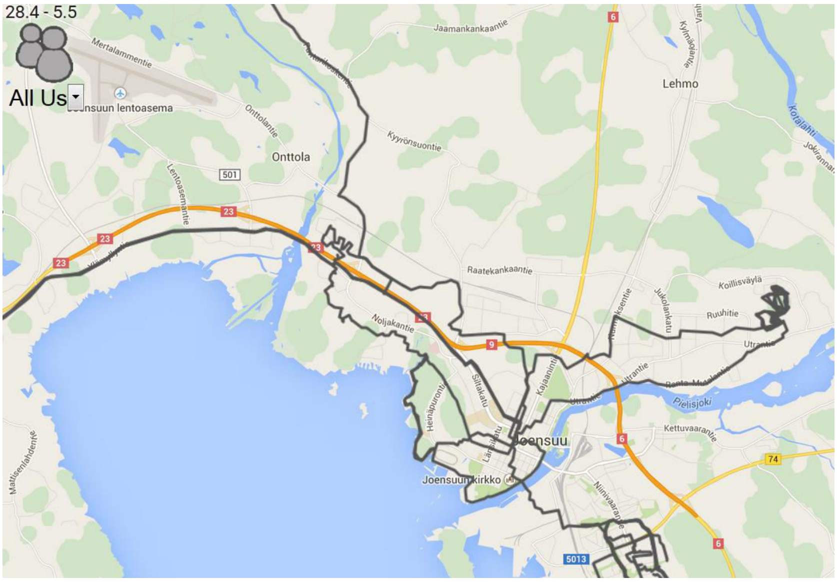


Figure 4. Collection of trajectories in the Mopsi system uploaded by all users in Joensuu area within single week.

Each of the trajectories can be analyzed. In the 'analyze view' as shown in Fig. 5, the Mopsi system shows segments of the trajectory including stop points, movement type classification and speed and altitude graphs.

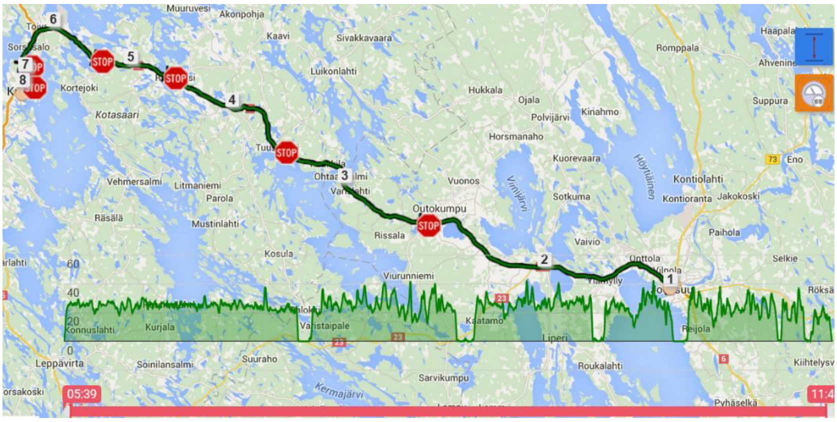


Figure 5. Analyzed GPS trajectory in the Mopsi system.

3 Processing and Analysis of GPS Trajectories

3.1 MOTIVATION

Advancing mobile phone technologies results in the higher availability of mobile devices and, in particular, higher availability of mobile devices with GPS. It is common that mobile devices are connected to the Internet. That results in the possibility of recording and sharing geotagged photos, tracking outdoor sports – for example, cycling and jogging – to peer users without use of specialized devices [23].

A large amount of location data coming to the server from mobile applications is difficult to efficiently store [21]. It is also a challenge to process, analyze and display the data to users in real time. Therefore, the main operations that need to be carefully designed in any location-based application are storage, retrieval and the visualization of location data. There are various types of location-based applications that need to process large amount of such data, for example tourist information [4], ride sharing, health monitoring [5], recommending points of interest [80][III] or sports tracking. Companies can manage their geographical information in real-time [58] and track the movement of their own vehicles in order to solve problems such as fleet management [42] or traffic congestion [59]. Nevertheless, access and visualization on maps that have large amounts of data is time consuming. For example, GPS trajectories shown on the map in Fig. 6 consist of thousands of points. There are several systems that address these problems. Visualization is carried out on mobile devices [29][48], on the web [2][5][95] or on specially designed separate applications [4][41].

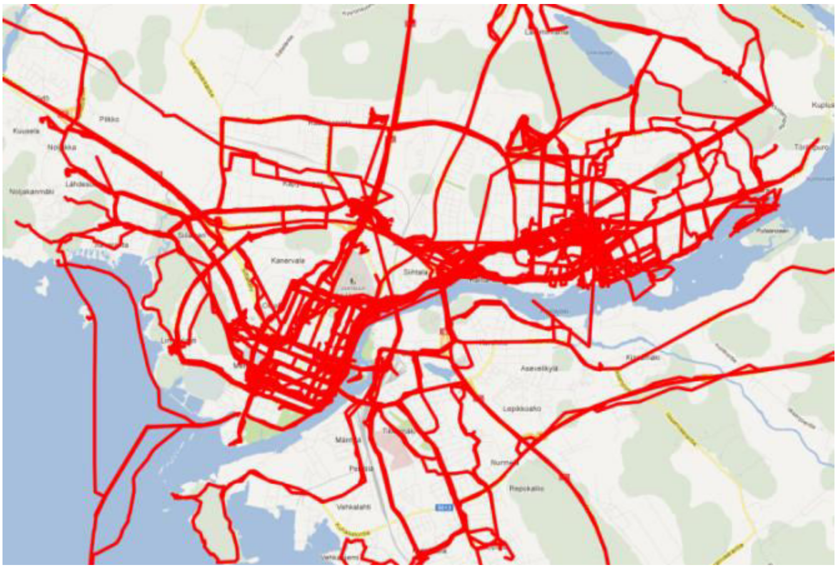


Figure 6. *Example of a user's GPS trajectory collection.*

We have designed a complete system for the storage, retrieval and visualization of trajectories that tackles the aforementioned problems. In addition, the system outperforms other existing real-time web based systems, such as GMapGIS, GPS Visualizer and Google Earth. None of these systems offer a possibility to plot trajectories consisting of thousands of points on a map. Moreover, the two first systems process the data slowly and are memory inefficient, often causing browsers to stop responding when attempting to plot several trajectories with approximately 10,000 points. Google Earth warns that processing may take longer than usual when plotting trajectories of over 2,000 points and say that the software responds slowly when the trajectory is plotted.

There exist approaches that aim to minimize the amount of data displayed by combining trajectories into trails that result from the approximation and interpolation of all overlapping trajectories [60]. Conversely, our solution displays all the recorded GPS trajectories matching the user's query in real-time by reducing the number of points being plotted but without any analysis and interpretation of the semantic meaning of the trajectories. This is achieved by applying a fast multi-resolution polygonal approximation algorithm as described in [20]. The algorithm achieves better approximation results than previous competitive methods. To speed up the visualization process, we apply a bounding box solution to the reduced tracks, so that only points that are visible to the user are plotted on map. The main goal was to create a system that can store and display a large amount of location data in form of GPS trajectories in real-time. A similar system, StarTrack, has been described in [5] and [39]. The system is the closest to the one we developed as it can handle up to 10,000 GPS trajectories. However, it does not address the problem of displaying trajectories on a map in real-time nor does it attempt to detect movement type. In addition, the StarTrack system was not tested on real-life GPS trajectories.

Similarly, as [39] we designed a system that handles GPS trajectories from the time they are sent to server by, in our case, mobile applications. The acquisition, storage, retrieval and visualization workflow of the system is illustrated in Fig. 7.

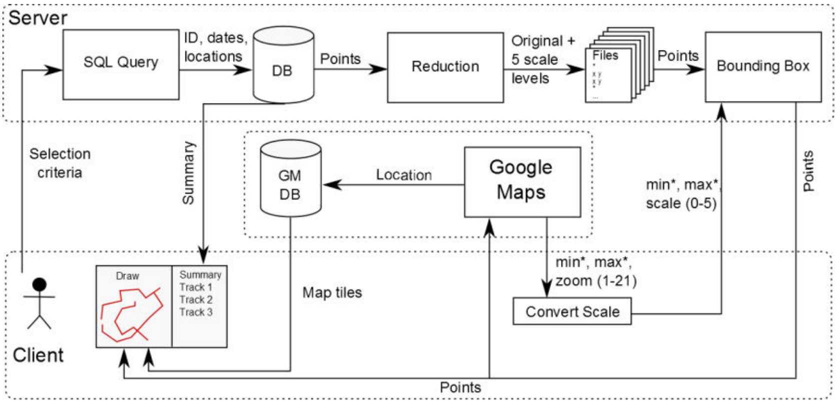


Figure 7. *Workflow of the GPS trajectories visualization system.*

After the trajectories are displayed user can analyze them further. For that purpose, various analysis algorithms were developed. For example, the algorithm described in [57] analyzes the similarity of the trajectories. Our system offers, in addition, a segmentation and classification of the segments of trajectories [II].

In addition, we provide basic statistics about each trajectory such as altitude and speed profiles.

The GPS data only captures features such as speed, distance, time and, obviously, location. The data cannot be straightforwardly used to conclude the semantic meaning of the user's activity [37]. Our system is able to detect movement type using only GPS data. We do not use the accelerometer as many competitive systems do, for example the system in [61], that aims to find movement type of GPS trajectories. In [64], the comparison of different combinations of GPS, GSM, Wi-Fi and the accelerometer for travel mode detection is available. The study concluded that the most useful for the travel mode detection task is one that possesses a combination of GPS and accelerometer data. The study mentions also other types of data used for the task, such as call detail records [82] and cellular network positioning data. Nevertheless, the latter types of data have not performed well [64].

The system we designed uses, however, only raw GPS data. The positive side of this is that no specialized devices, such as the accelerometer, are needed to collect data. We work with the data that can be collected with a mobile phone that has embedded GPS. However, in our system, we analyze various characteristics and features of the GPS trajectory. The simplest approach to determine the travel method of a user is to measure speed [14][77] of movement, which is a trivial task having location points with a timestamp in the GPS trajectory. Some transport modes though, such as cycling and running, are difficult to differentiate by speed thresholds alone. For that purpose, more complex solutions, such as fuzzy logic [87], neural networks [34] and hidden Markov model [64][94] have been considered.

Our goal is to detect five typical movement types: stationary, walk, run, bicycle and motor vehicle. The approach is to find segments of GPS trajectories that have similar features and apply classifier on such a segmented track as shown in Fig. 8.

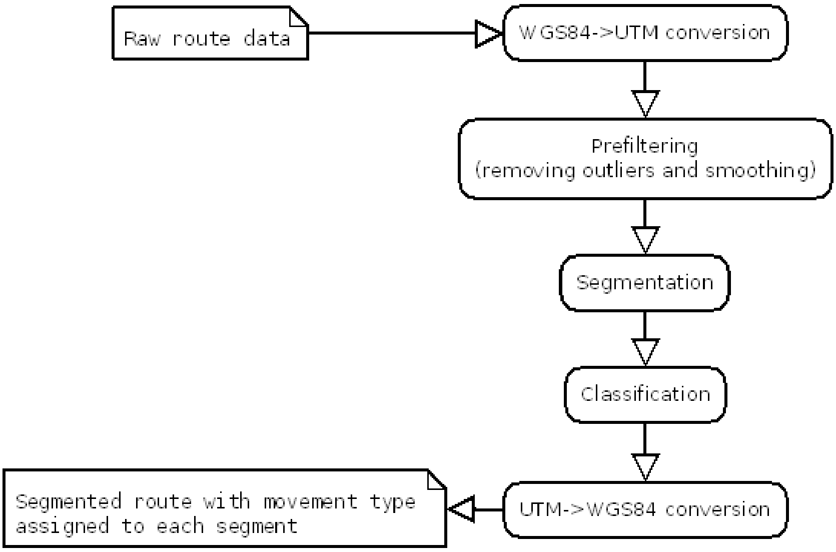


Figure 8. *Workflow of the segmentation and classification process.*

As GPS data has many inaccuracies, the system starts with preprocessing the trajectory. At this step, all potential outliers are identified and removed by checking their speed consistency. The trajectory is also smoothed. The preprocessed trajectory is input to the segmentation algorithm. A set of basic features, such as speed, acceleration, time, direction and distance are extracted for each segment. The movement type is then detected using a second order Markov model.

3.2 DATA ACQUISITION AND STORAGE

The system collects trajectories using mobile application. The application is available for major operating systems: Android, WindowsPhone, iOS and Symbian. The mobile application records location and timestamp at predefined intervals (usually from 1 to 4 seconds). The mobile application is only responsible for the collection and display of data. Contrary to [8], the data is processed entirely on the server for the reason that all computations can be carried out faster on the server. If an Internet connection is available, the data is immediately saved to the database on the server. Otherwise the data is buffered on the device.

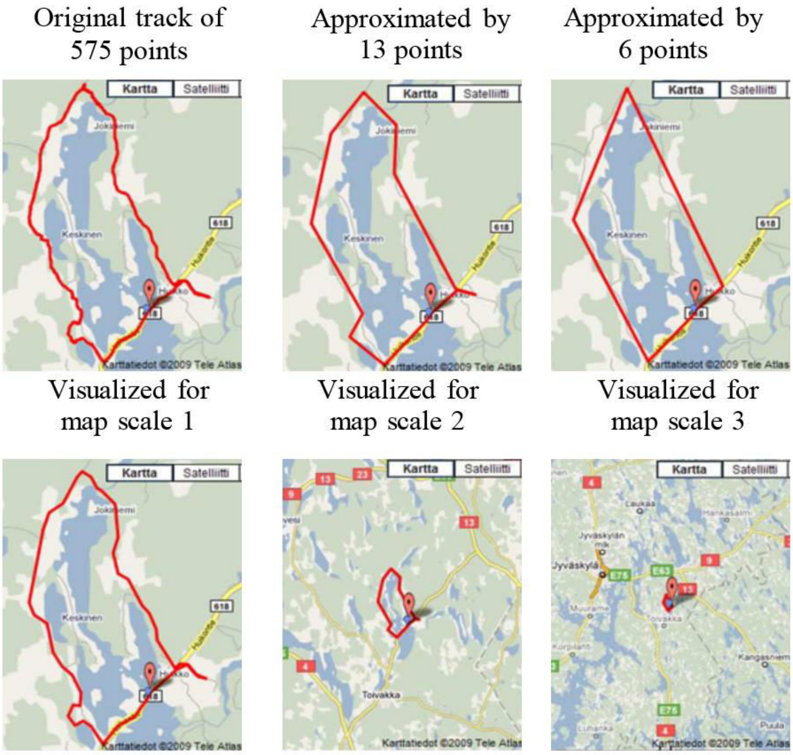
Trajectories are at first saved as individual points in the database. The GPS trajectory objects are created and updated in real-time when new points arrive to the server. Each trajectory is associated with several basic statistics such as start and end time, bounding box, number of points, segments and movement type. The segmentation and movement type classification of trajectories are described in more detail later on in this chapter. The trajectory is stored in its original form, i.e. the sequence of time-stamped location points, and also in a simplified form that contains reduced number of points. The approximated trajectories are computed for five different zoom levels of map [20]. This allows for faster visualization of tracks on the map, as the number of points drawn is significantly smaller. Nevertheless, the GPS trajectory shape is not distorted. The approximated tracks and analyses are performed immediately when the original trajectory is uploaded.

Trajectories are uploaded and constantly analyzed as they are uploaded to server. The statistics must be updated each time new points of the trajectory come into play. To ensure that the statistics are updated in real-time, there is a constant process running on server and periodically checking (every minute) if any trajectory in the system has received new points. Whenever new points are uploaded to server, the process decides whether a new trajectory should be created or the points should be merged instead with one of the existing trajectories. The existing trajectories are updated in case new tracking points belonging to an older trajectory are received after a significant delay that has been caused, for example, by a poor Internet connection.

3.3 VISUALIZATION

The original trajectories recorded in the system contain more data than needed for visualization. The full data, though, is required for analysis, and therefore complete trajectories are stored.

However, in the rendering process for the web browser, a reduced number of points are sufficient to represent the shape of the trajectory to the user. To create approximated trajectories that preserve the original shape of the trajectory, the system applies a multi-resolution polygonal approximation algorithm as described in [20]. The algorithm is applied to every received trajectory. It approximates the trajectory for five different map scales. The algorithm has O(n) time complexity and the results are stored in the database in order to avoid the repetitive execution of the algorithm for the same input trajectory when the trajectory is again displayed. Figure 9 shows an example of the original and approximated tracks.



Original track of 575 points

Approximated by 13 points

Approximated by 6 points

Visualized for map scale 1

Visualized for map scale 2

Visualized for map scale 3

Figure 9. Visualization of the original and approximated tracks.

The original track contains 575 points and is approximated in different map scales with 44, 13, and 6 points respectively. A suitable approximation for error tolerance is selected for each map scale, so that the visualization quality is not affected by the approximation, although rendering time is reduced significantly.

In a further attempt to reduce the amount of data that needs to be displayed on the map, we apply a bounding box, so that only parts of trajectories visible at the moment on the user's screen are plotted on the map. Therefore, we only select the points that user will see while using the current map scale and location (the *bounding box* of the map) at the moment of query. In addition, the system also draws points that are outside the bounding box but within immediate neighborhood (50% extension of screen size). In this way, we prepare for panning and zooming by the user. The way the bounding box works is shown in Fig. 11. Here, the bounding box is implemented as a function that makes the coordinates of north, south, east and west of the map visible on the screen. The map scale is also passed so that points from the correct approximation can be selected. The function is applied to every track and for every point it checks if the point lies within the bounding box. The time complexity of the bounding box is linear and is entirely computed on server.

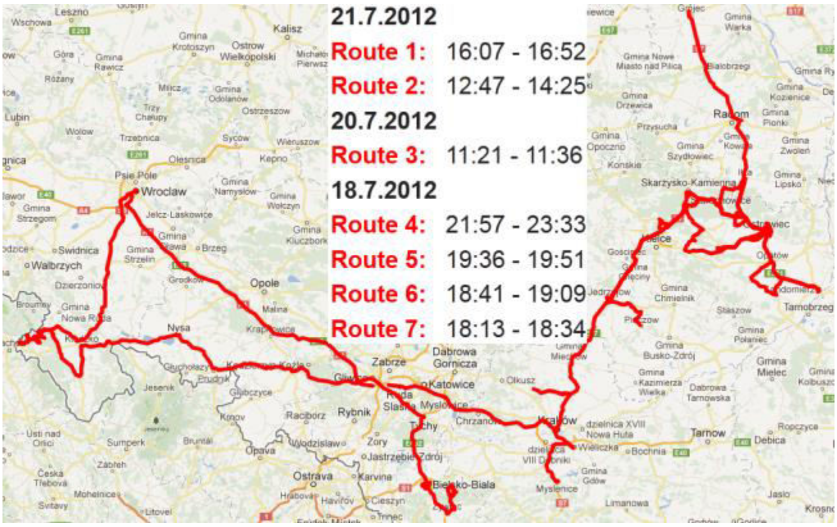


Figure 10. *Visualization of sample GPS trajectories.*

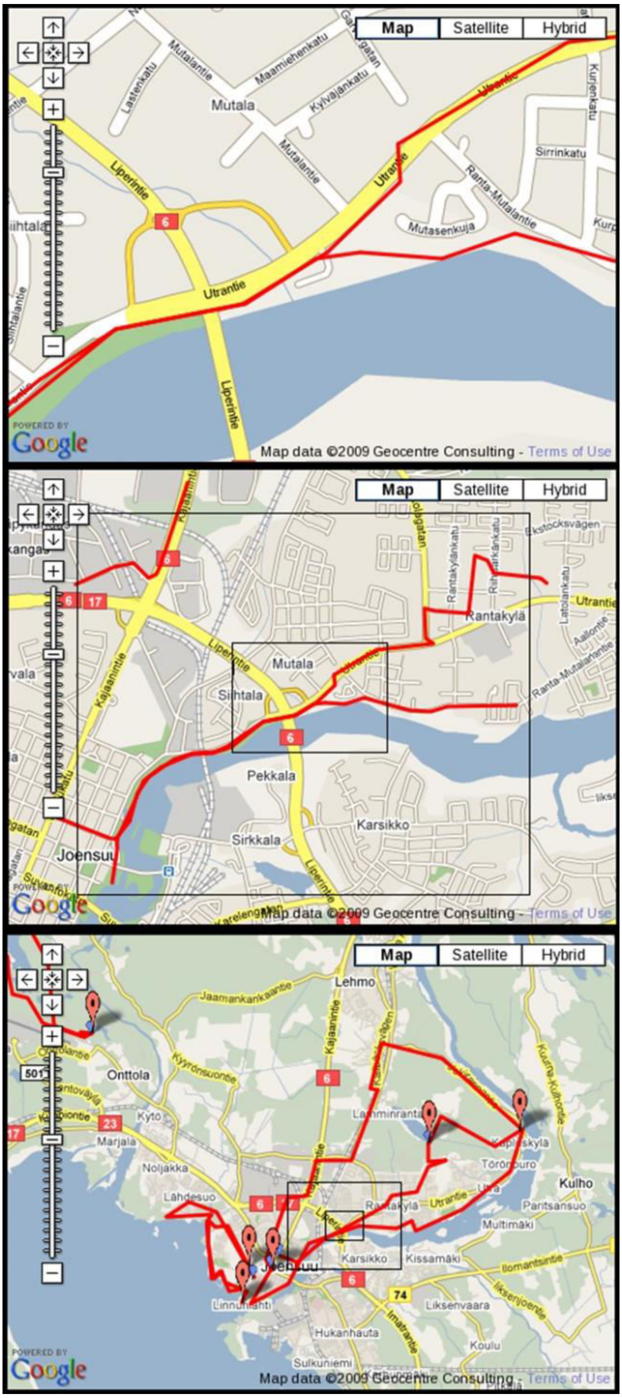


Figure 11. *Bounding box. Top – what the user can see, middle – what is actually plotted on the map, bottom – all trajectories matching selection criteria.*

For displaying trajectories on map, the system uses Google Maps API. However, the displayed map can be changed. The user can select, for instance, OpenStreetMap to be shown as an overlay on Google Maps. Similarly, customized maps can be shown as, for example orienteering maps. The GPS trajectories can be browsed according to time. The results of time query are displayed as shown in Fig. 10.

To evaluate the system, we measure the time spent between sending requests to the system and presenting the results to the user. In the measurements, the time needed for data transfer is ignored. Nevertheless, the system is designed in such a way that the data transfer is minimized.

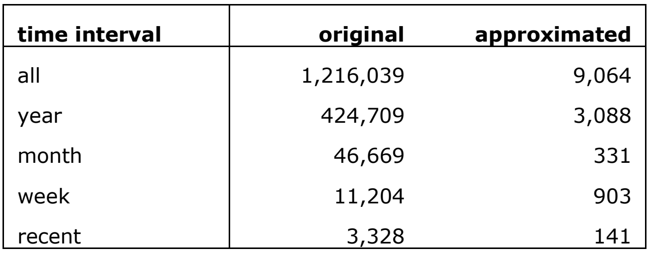
The experiment has been conducted on the dataset that is summarized in Table 1.

Table 1. *Summary of data used in experiments.*

User	Tracks	Points	Length (km)	Duration (h)
Pasi	784	1,216,039	8,535	669
Karol	650	1,015,939	9,655	442
Radu	429	613,684	4,604	188

The original trajectories consist of a large number of points. As shown in Table 2, there are users in the dataset that have over one million points.

Table 2. *Number of points in GPS trajectories collected by user Pasi.*



time interval	original	approximated
all	1,216,039	9,064
year	424,709	3,088
month	46,669	331
week	11,204	903
recent	3,328	141

This demonstrates the need to approximate the trajectories, as none of the browsers can handle such a large amount of data [23]. In tests, the zoom level is selected so that all the trajectories are visible on the map. We measured the durations of the three stages of the visualization process: the querying database, the computing bounding box and visualization in browsers. The times measured for users taking the tests are shown in Fig. 12. Results show that the time needed for showing all the tracks of the user with the largest collection is about 2.5 seconds. Querying the data takes up most of the time, as shown in Fig. 13. Calculating the bounding box is a fast process that additionally speeds up drawing trajectories on map, taking up only 14% of the allocated time.

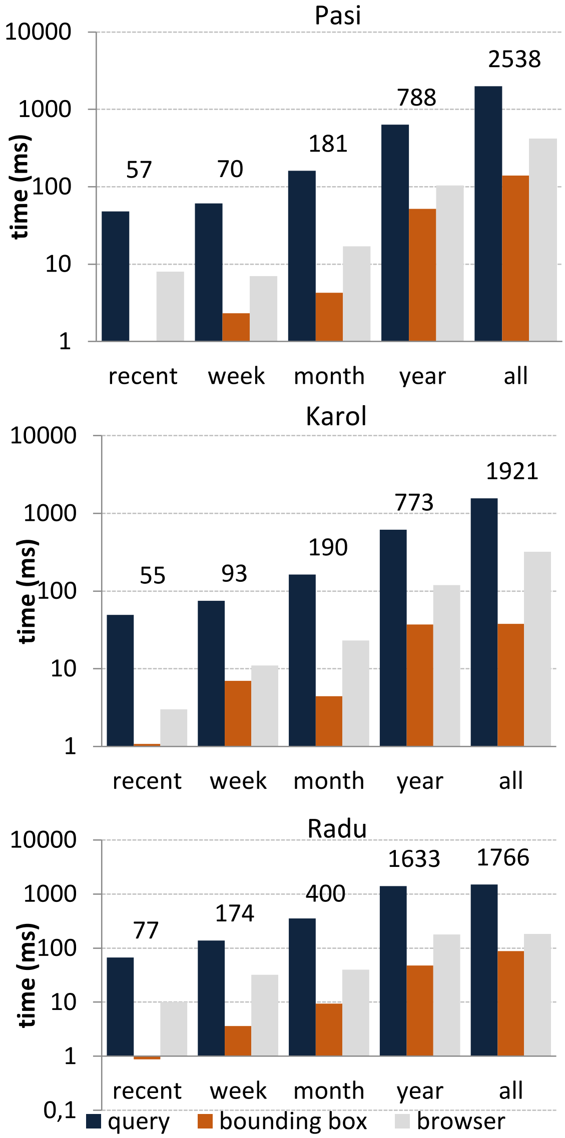


Figure 12. *Time needed to display tracks in a selected period for three test users.*

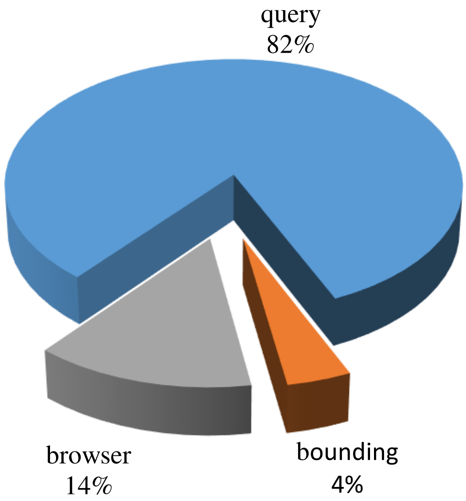


Figure 13. *Average time (in percent) spent in each of the three phases of the visualization process.*

The approximation algorithm is necessary to reduce the number of points displayed. Without it, it is not possible to display all the tracks because the web browser would either stop responding or crash. The number of points that browsers can handle depends on the available resources. Displaying thousands of points slows down web browsers. Even if a browser can display all the points, the time needed for the process increases significantly with the increase of the number of trajectories visualized. Table 3 shows the sizes of files that contain the trajectory collection of one user.

Table 3. *Size of files (in bytes) with original and approximated tracks for user Karol.*

time interval	original	approximated
week	14.000	148
month	346.000	2280
year	4.056.000	69.000
all	11.595.000	129.000

Experiments show that applying the bounding box decreases the time needed to draw tracks on the map. Figure 13 shows a sample case from the experiments. To test this, the same set of tracks was requested at the same zoom level, but the map was focused in two different places: Finland and Poland. In Finland, the collection of tracks was big, whereas in Poland only a few tracks were recorded. As the bounding box solution is applied, not all the tracks have to be displayed. The time it took to show the smaller number of tracks (Poland area) was significantly shorter than the larger (Finland area). Figure 14 also shows how reducing the number of points affects display time.

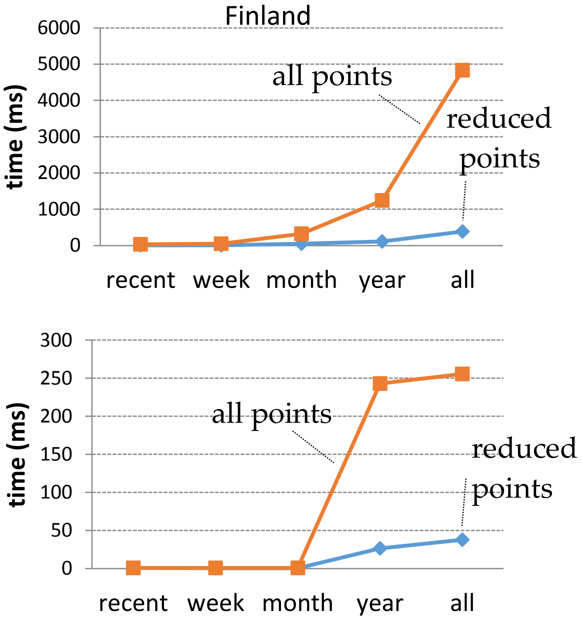


Figure 14. *Example of querying the same track collection with map set to the same zoom level when focused in Finland with large collection of tracks (top) and Poland with small collection of tracks (bottom).*

In comparison with the existing web based systems for visualizing GPS trajectories, for instance GMapGIS and GPS Visualizer, our system can handle the display of data consisting of significantly more points. Moreover, joint application of the trajectory reduction and the bounding box makes the system interactive even with large number of trajectories shown at the same time. This makes it different to any other existing web based trajectory visualization system. For example, attempt to draw 800 trajectories consisting of over 1,200,000 points in Google Maps causes that the browser stops responding and uses over 10% of CPU and 1GB of memory. According to our experiments, GPS

Visualizer causes the same behavior of browser when plotting several trajectories with approximately 10,000 points only.

Furthermore, algorithms used in Google Maps could be further optimized. Speed of clustering of markers plotted on the map can be improved significantly by applying clustering proposed in [65]. Data size of 1K, 10K, 100K, 1M requires 0.3 s, 1.7 s, 20 s and 229 s by Google Maps clustering, and 0.23 s, 0.34 s, 0.64 s, 2.4 s with proposed clustering, to be displayed on the map.

3.4. FILTERING

When the trajectories are displayed to the user, the system gives the opportunity to analyze each of the trajectories and check detailed statistics such as speed and elevation, as well as to search for similar trajectories. Part of this analysis is the classification of movement type. Our solution is a three-step process that consists of preprocessing, segmentation and the actual classification of each segment to one of the five movement types as shown in Fig. 8.

The trajectories are stored in a relational database as described in Chapter 2. For each point, we store several characteristics describing the GPS signal at the time of taking measurement. Nevertheless, only location and timestamp are needed for our filtering, segmentation and classification.

There is a possibility that the GPS receiver provides inaccurate location data. It can be caused by various factors, such as weather conditions (clouds), surrounding environment (high buildings, tunnels), and interference with other devices or bad quality receivers. Fig. 15 shows an example of such GPS inaccuracies in trajectory collection such as outlier points and zigzag trajectories.

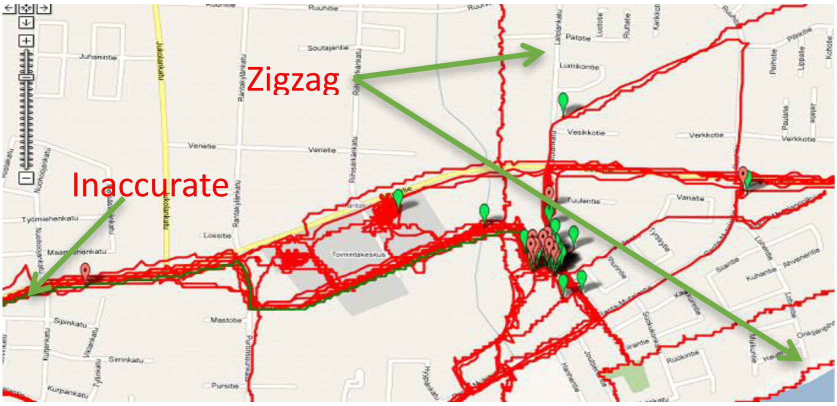


Figure 15. Example of GPS inaccuracy in tracking.

Location errors affect the results of trajectory analyses. Even speed calculations can be strongly affected by outlier points. The inaccuracies also affect our segmentation and classification algorithm. Filtering needs to be applied whenever GPS data is noisy and whenever it is required to extract data about movement such as speed or direction. Therefore, we filter trajectories by default.

Different approaches have been applied to preprocessing of GPS trajectories [47]. Our approach is to remove outliers and filter the trajectory points using an assumption that speed is consistent. The advantage of this approach is that it relies solely on GPS data and does not require any prior information, such as road network, contrary to other algorithms [47]. Moreover, as pointed out in [20], filtering algorithms rely on sets of parameters that are often difficult to estimate and may vary, even in different parts of the same GPS trajectory.

The first step in our filtering algorithm is to eliminate points with impossibly high speed and speed variance. Such points are usually easy to identify, as shown in Fig. 16 where we can observe a movement to the other side of the river in one second. In addition, according to the map there is no bridge, therefore such movement is impossible for a human. To identify such outliers, we calculate speed between each pair of adjacent points in the trajectory and remove points that have a higher speed than a given threshold. When such points are removed, the algorithm searches for points with abnormal acceleration and unusual changes of direction. This is used to identify potential additional outliers that were missed when the speed variance threshold criterion was applied. Points are denoted as outliers if the acceleration exceeds a maximum threshold value, or if the acceleration is high and the direction of movement rapidly changes.

The second step in the algorithm is to smoothen trajectory. As shown in Fig. 15, GPS errors may cause a recorded trajectory to zigzag even if the points of the original trajectory form a straight line. For this reason, we apply a smoothing algorithm to the trajectory after the outliers have been removed in the first step. The algorithm divides the trajectory into short 2-minute segments. Each of them is smoothed separately using Tikhonov's regularization with a speed-smooth regularized term. The smoothed trajectory is a result of averaging smoothing results for each 2-minute segment. The results of this smoothing effect are shown in Fig. 17.

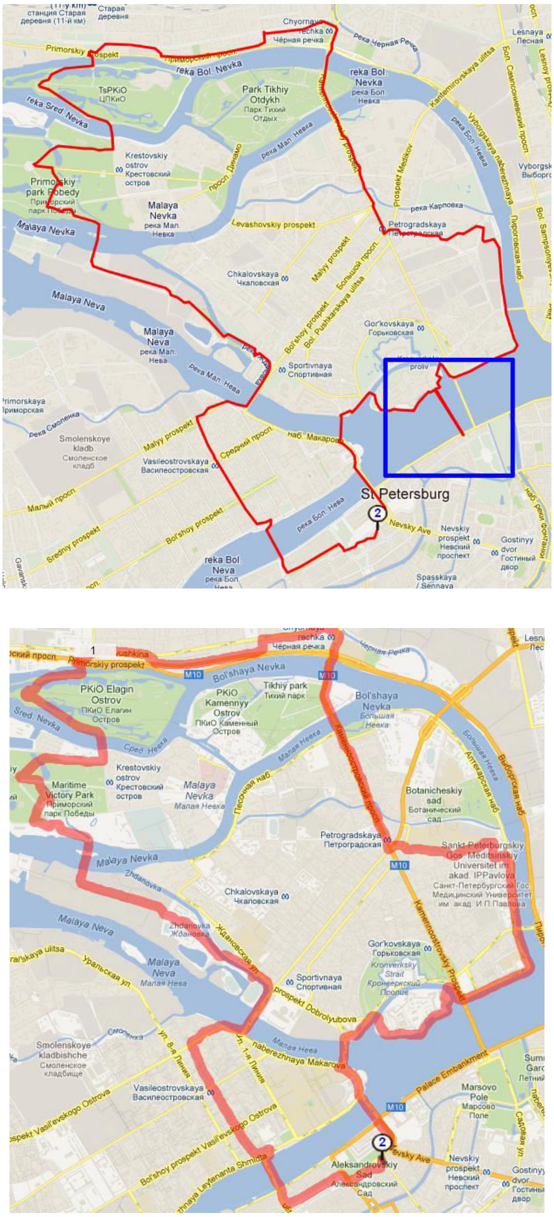


Figure 16. *GPS trajectory with impossible movement example (part in blue frame) before and after filtering.*

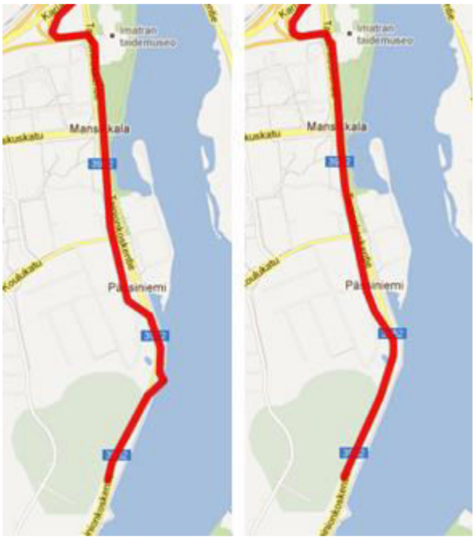


Figure 17. *A trajectory before (left) and after smoothing (right).*

3.5 SEGMENTATION

Having filtered the trajectory without GPS inaccuracies, we proceed to further analyze the trajectory. The motivation for the segmentation of the trajectory before aiming at movement type classification is illustrated in Fig. 18 and 19. The first example in Fig. 18 shows a GPS trajectory that was recorded during interval running training with two slower jogging periods in between. The second example in Fig. 19 shows three fast downhill skiing segments divided by queuing and time spent on ski lift. Although it is impossible to deduce all these activities from basic GPS data, segmentation can help to get better classification results and to find segments that have similar characteristics.

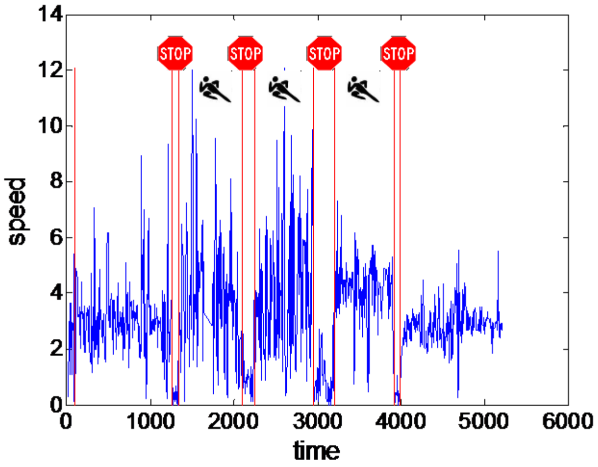


Figure 18. *Non-trivial examples of movement type analysis: interval training.*

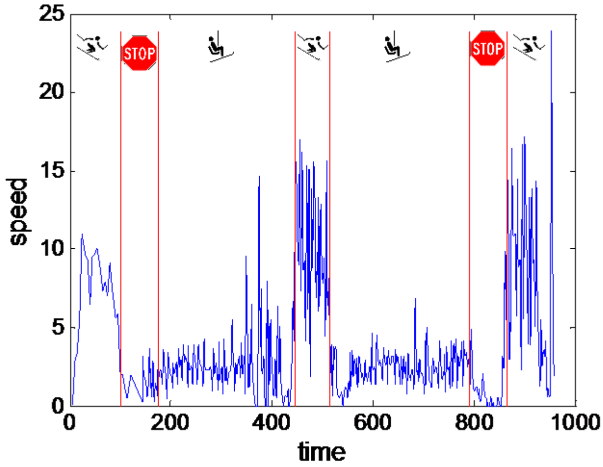


Figure 19. *Non-trivial examples of movement type analysis: quality downhill skiing time and time spent in queue and in ski lift.*

The algorithm divides the trajectory into segments based on similar speed. The number of segments is automatically determined. However, the user can also provide the number of segments. To segment the trajectory, we build a cost matrix for the connection costs between the location points. The cost matrix is based on a speed criterion assuming that speed variance within a segment is small. The sum of speed variances for all segments is then minimized.

Let us consider a trajectory that consists of n points. The points form a set $P=(p_1, p_2,\ldots,p_n)$, where $p_i=(x_i, y_i, t_i)$. The corresponding speed set is $V=(v_1, v_2,\ldots,v_{n-1})$. For a given number of segments m, we define a cost function that minimizes the sum of the inner speed variance in all the segments:

$$\mathrm{f} = \sum_{j=1}^{m} \sigma(V_j) \cdot (t_{i_{j+1}} - t_{i_j}) \tag{1}$$

where i_j and i_{j+1} are the indexes of the start and end points of the segment j, V_j is the set of speeds of segment j, and $\sigma(V_j)$ is the speed variance of the segment j. Our experiments have shown that the proposed cost function works better than the mean square error, which has difficulty in detecting walking segments of low speed.

The minimization process is solved by dynamic programming in $O(n^2m)$ time and $O(nm)$ space because the speed variance can be calculated in $O(1)$ time by using the pre-calculated accumulated sums. Optimization is carried out as follows:

$$\begin{gathered} \mathrm{D(s,r)} = min\big(D(c, r-1) + \sigma_c^s(t_s - t_c)\big), c = 1 \ldots s-1 \\ \mathrm{A(s,r)} = argmin_c(D(c, r-1) + \sigma_c^s(t_s - t_c)) \end{gathered} \tag{2}$$

where $s = 0\ldots n$, $r = 0\ldots m$ with an initial condition $D(0, 0) = 0$, $A(s, r)$ is the index for backtracking.

The number of segments m_0 is determined by

$$m_0 = argmin_i\big(D(n,i) + \lambda_1 i + \lambda_2(t_n - t_1)\big), i = 1 \ldots m \tag{3}$$

where λ_1, λ_2 are regularization parameters.

3.6 MOVEMENT TYPE DETECTION

Our goal is to classify each segment as stationary, walk, run, bicycle or motor vehicle. We calculate several features such as speed, acceleration, time, direction and distance. However, training a classifier on these specific features might not be accurate as many features overlap with different movement types [94]. Instead, we first perform a soft classification of each segment as stationary, walk, run, bicycle or motor vehicle using *a priori* probabilities shown in Fig. 20.

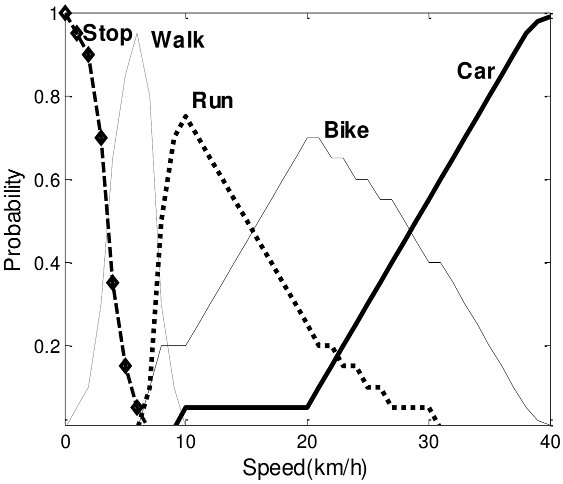
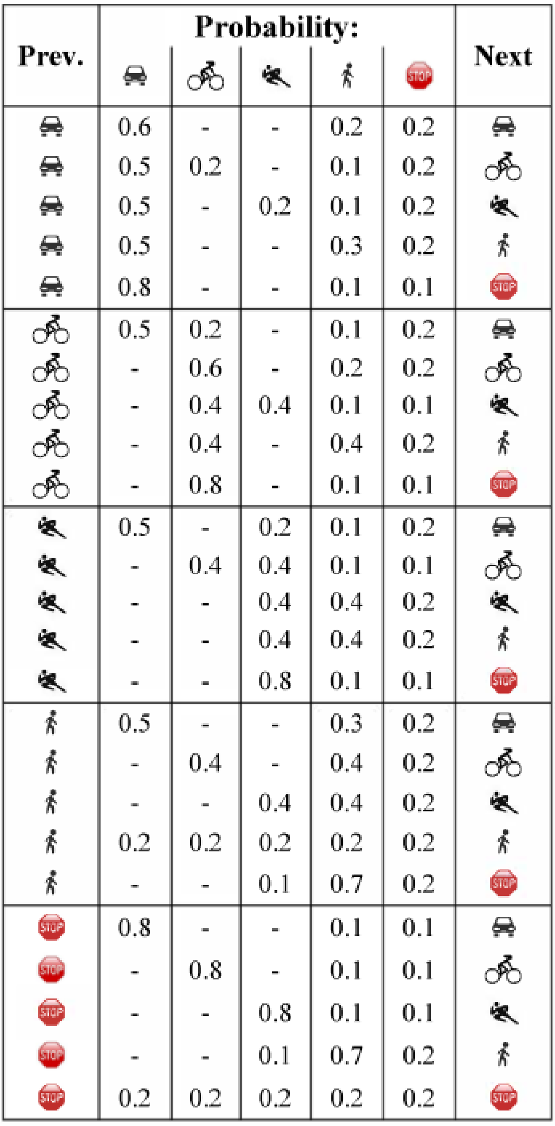


Figure 20. A priori *probabilities for soft classification of the trajectory segments.*

The first order hidden Markov model (HMM) has been used to exploit the correlations between neighboring segments in [64]. In this model, the hidden states represent each of the movement types and the observed data are the features of each segment. We extend this to a second-order HMM to exploit the correlation between both the previous and the next segment. The state transition matrix is empirically constructed, as in Fig. 21, but could also be optimized via a training process in further stages of the application. For the cost function, we use function *f*, which is defined as follows:

$$f = \prod_{i=1}^{M} P(m_i|X_i, m_{i-1}, m_{i+1}) \tag{4}$$

where m_i = *{stop, walk, run, bicycle, motor vehicle}* is the state of segment *i*, X_i is its feature vector, and m_{i-1}, m_{i+1} are the states of the previous and the next segment. Thus, the probability that a segment has a hidden state m_i depends on the previous state, the next state and its feature vector. After maximizing the function shown in Eq. 4, we determine the most likely sequence of the hidden state m_0, m_1… m_M. The sequence represents the most likely movement type and potential transitions between different movement types.



Prev.	Probability:					Next
	🚗	🚲	⛷	🚶	STOP	
🚗	0.6	-	-	0.2	0.2	🚗
🚗	0.5	0.2	-	0.1	0.2	🚲
🚗	0.5	-	0.2	0.1	0.2	⛷
🚗	0.5	-	-	0.3	0.2	🚶
🚗	0.8	-	-	0.1	0.1	STOP
🚲	0.5	0.2	-	0.1	0.2	🚗
🚲	-	0.6	-	0.2	0.2	🚲
🚲	-	0.4	0.4	0.1	0.1	⛷
🚲	-	0.4	-	0.4	0.2	🚶
🚲	-	0.8	-	0.1	0.1	STOP
⛷	0.5	-	0.2	0.1	0.2	🚗
⛷	-	0.4	0.4	0.1	0.1	🚲
⛷	-	-	0.4	0.4	0.2	⛷
⛷	-	-	0.4	0.4	0.2	🚶
⛷	-	-	0.8	0.1	0.1	STOP
🚶	0.5	-	-	0.3	0.2	🚗
🚶	-	0.4	-	0.4	0.2	🚲
🚶	-	-	0.4	0.4	0.2	⛷
🚶	0.2	0.2	0.2	0.2	0.2	🚶
🚶	-	-	0.1	0.7	0.2	STOP
STOP	0.8	-	-	0.1	0.1	🚗
STOP	-	0.8	-	0.1	0.1	🚲
STOP	-	-	0.8	0.1	0.1	⛷
STOP	-	-	0.1	0.7	0.2	🚶
STOP	0.2	0.2	0.2	0.2	0.2	STOP

Figure 21. *Probability matrix for second order hidden Markov model.*

Assuming that the feature vector X_i is uncorrelated with m_{i-1} and m_{i+1}, this cost function can be converted by applying the Bayesian inference. The inference algorithm for classifying the movement type of a segment maximizes the cost function f. We first apply the distribution rule and Bayesian inference to the probability in the cost function from Eq. 4.

$$P(m_i|X_i, m_{i-1}, m_{i+1}) = P(m_i|m_{i-1}, m_{i+1}, X_i) \\ = \frac{P(m_{i-1}, m_{i+1}, X_i|m_i)P(m_i)}{P(m_{i-1}, m_{i+1}, X_i)} \tag{5}$$

We assume that the feature X_i is independent of m_{i-1} and m_{i+1} so we decompose probabilities $P(m_{i-1},m_{i+1}, X_i \mid m_i)$ and $P(m_{i-1},m_{i+1}, X_i)$:

$$P(X_i|m_i)P(m_{i-1}, m_{i+1}|m_i)\frac{P(m_i)}{P(m_{i-1}, m_{i+1})P(X_i)} \tag{6}$$

We apply the Bayesian inference to $P(X_i|m_i)$ and $P(m_{i-1}, m_{i+1}|m_i)$ so that Eq. 6 becomes

$$\frac{P(m_i|X_i)P(X_i)}{P(m_i)}\frac{P(m_i|m_{i-1}, m_{i+1})P(m_{i-1}, m_{i+1})}{P(m_i)}\frac{P(m_i)}{P(m_{i-1}, m_{i+1})P(X_i)} \\ = \frac{P(m_i|m_{i-1}, m_{i+1})P(m_i|X_i)}{P(m_i)} \tag{7}$$

Using Eq. 7, the cost function takes the following form:

$$f = \prod_{i=1}^{M}\frac{P(m_i|m_{i-1}, m_{i+1})P(m_i|X_i)}{P(m_i)} \tag{8}$$

Because m_{i-1} and m_{i+1} are not known when considering m_i, we need to optimize the calculation of the cost function so we can use it in a sequential process by modifying $P(m_i|m_{i-1}, m_{i+1})$ so that $i \leq i+1$:

$$P(m_i|m_{i-1}, m_{i+1}) = \frac{P(m_{i+1}|m_{i-1}, m_i)P(m_i)}{P(\mathrm{m}_{i+1})} \tag{9}$$

The equation becomes:

$$f = \prod_{i=1}^{M}\frac{P(m_{i+1}|m_{i-1}, m_i)P(m_i)}{P(m_{i+1})}\frac{P(m_i|X_i)}{P(m_i)} \\ f = \prod_{i=1}^{M}\frac{P(m_{i+1}|m_{i-1}, m_i)P(m_i|X_i)}{P(m_{i+1})} \tag{10}$$

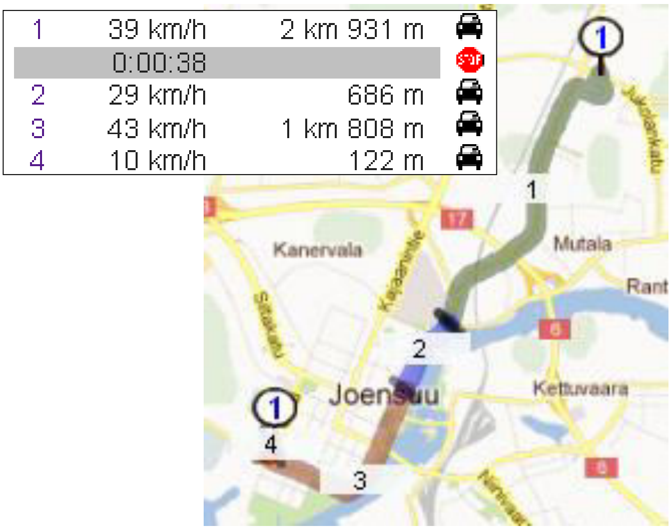
The cost function can finally be written as:

$$f = \prod_{i=1}^{M} \frac{P(m_{i+2}|m_i, m_{i+1})P(m_{i+1}|X_{i+1})}{P(m_{i+2})} \qquad (11)$$

where $P(m_{i+2}|m_i, m_{i+1})$, $P(m_{i+1}|X_{i+1})$ and $P(m_{i+2})$ are all given as prior information.

Dynamic programming is employed for maximizing the cost function from Eq. 11 in a similar manner to the Viterbi algorithm, which has been used for the first order HMM.

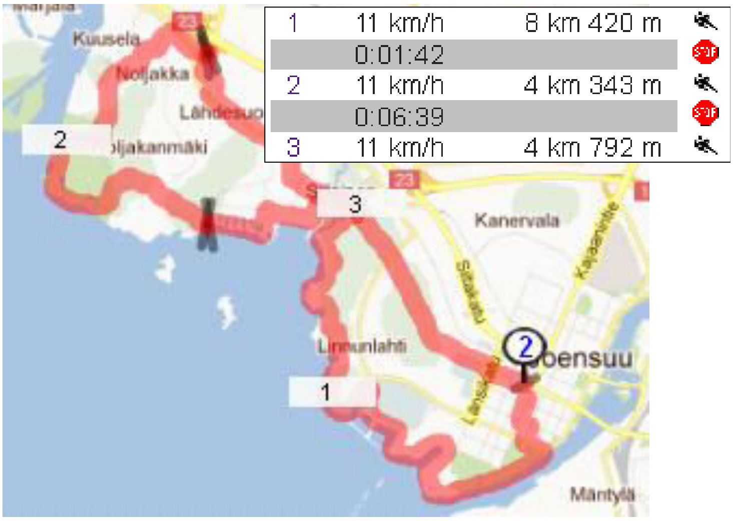
The filtering, segmentation and classification algorithms have been tested with the Mopsi data. The algorithms are triggered by a user when he or she wishes to see the detailed statistics of the GPS trajectory.



1	39 km/h	2 km 931 m	🚗
	0:00:38		🛑
2	29 km/h	686 m	🚗
3	43 km/h	1 km 808 m	🚗
4	10 km/h	122 m	🚗

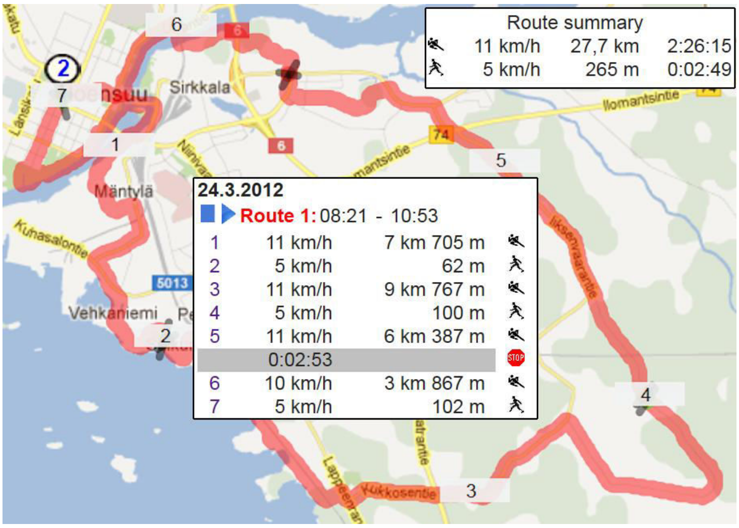
Figure 22. Segmentation of a car travelling with just one stop.

Fig. 22 shows a car travelling, including one stop at traffic lights. The segments demonstrate typical traffic flows in Joensuu. The last segment was recorded while looking for a parking place and is classified as a motor vehicle, despite its lower speed.



1	11 km/h	8 km 420 m	🏃
	0:01:42		STOP
2	11 km/h	4 km 343 m	🏃
	0:06:39		STOP
3	11 km/h	4 km 792 m	🏃

Figure 23. *Separating stop segments from running.*



Route summary

🏃	11 km/h	27,7 km	2:26:15
🚶	5 km/h	265 m	0:02:49

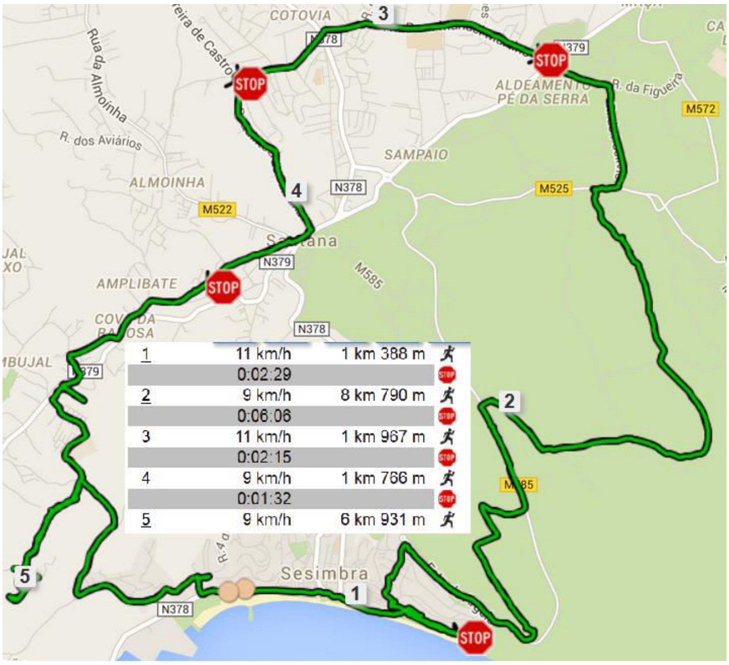
24.3.2012

Route 1: 08:21 - 10:53

1	11 km/h	7 km 705 m	🏃
2	5 km/h	62 m	🚶
3	11 km/h	9 km 767 m	🏃
4	5 km/h	100 m	🚶
5	11 km/h	6 km 387 m	🏃
	0:02:53		STOP
6	10 km/h	3 km 867 m	🏃
7	5 km/h	102 m	🚶

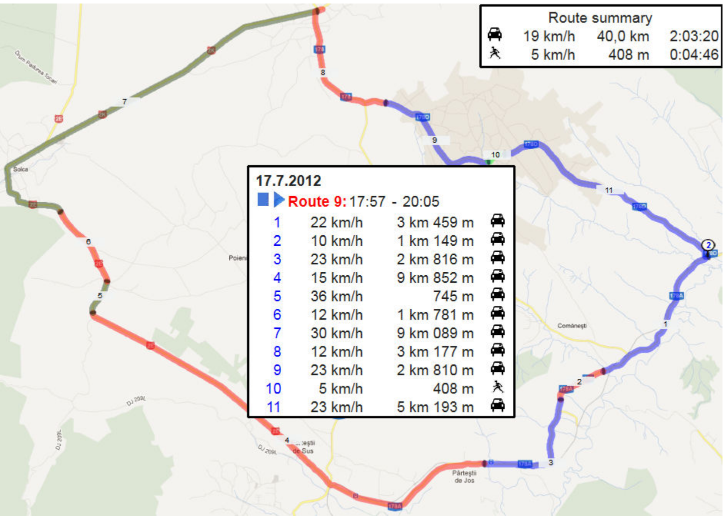
Figure 24. *Long-distance running.*

Fig. 23, Fig. 24 and Fig. 25 show sports exercises where it can be easily concluded that the user is running, judging by the speed of the moving segments, even though there are some short stops (segment 2 from Fig. 24) that are incorrectly classified as walking.



1	11 km/h	1 km 388 m	🏃
	0:02:29		STOP
2	9 km/h	8 km 790 m	🏃
	0:06:06		STOP
3	11 km/h	1 km 967 m	🏃
	0:02:15		STOP
4	9 km/h	1 km 766 m	🏃
	0:01:32		STOP
5	9 km/h	6 km 931 m	🏃

Figure 25. Interval training exercise.



Route summary

🚗	19 km/h	40,0 km	2:03:20
🚶	5 km/h	408 m	0:04:46

17.7.2012

Route 9: 17:57 - 20:05

1	22 km/h	3 km 459 m	🚗
2	10 km/h	1 km 149 m	🚗
3	23 km/h	2 km 816 m	🚗
4	15 km/h	9 km 852 m	🚗
5	36 km/h	745 m	🚗
6	12 km/h	1 km 781 m	🚗
7	30 km/h	9 km 089 m	🚗
8	12 km/h	3 km 177 m	🚗
9	23 km/h	2 km 810 m	🚗
10	5 km/h	408 m	🚶
11	23 km/h	5 km 193 m	🚗

Figure 26. Bicycle route classified as car.

Finally, most of the segments in the trajectory shown in Fig. 26 are incorrectly classified as a motor vehicle. Despite the accurate segmentation and the correct detection of the walking segment, the inaccuracies of the GPS signal and high top speed classify cycling as a motor vehicle segment. Classifying one segment as motor vehicle movement increases the probability of similar segments being classified as the same transportation mode.

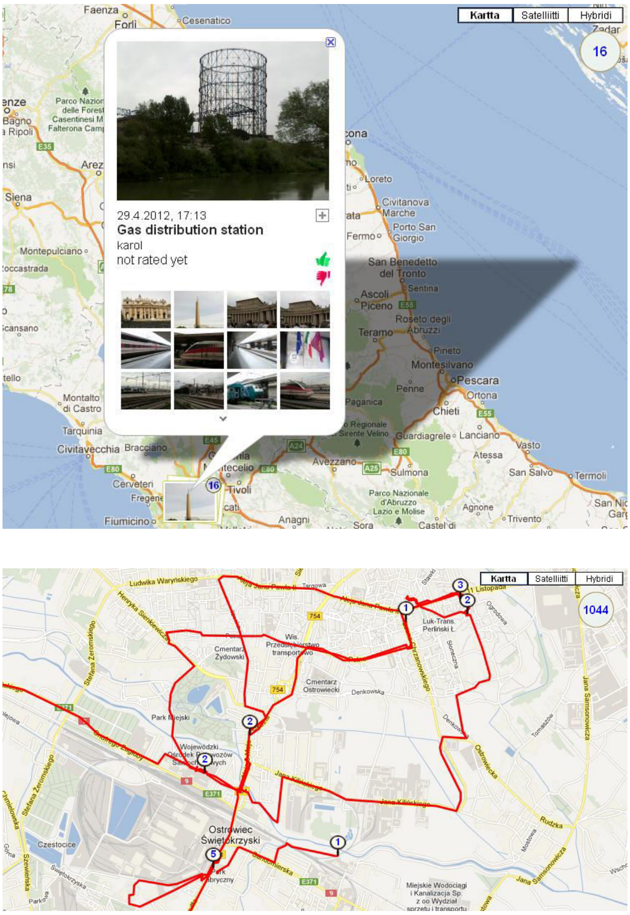
4 Context-aware Recommendation of Location Data

4.1 MOTIVATION

The high availability of mobile phones equipped with GPS and mobile Internet allows people to record their activities and share them with others [33]. The analysis of such data, including, for example, geotagged photos and GPS trajectories shown in Fig. 27, reveal patterns of user movements and information about points of interest in certain areas. The data is easily available as users are encouraged by their peers to collect it. Furthermore, the data can be used to recommend activities and places for new users to visit in the area.

Personalized recommendation systems are sometimes based on static user profiles, but more often on user actions in a system [13] and are widely used for recommending similar products in online stores, videos on YouTube, friends and groups on Facebook, and advertisements targeted to a specific audience. Recommendation systems are in scope of interest of research institutes and companies [1].

We next review the recommendation system implemented in Mopsi [80][III], which is based on the four aspects of relevance: content, time, location and social network as identified in [30]. The system recommends that items from a user generated location data collection that includes geotagged photos, GPS trajectories and services as described in Chapter 2. The goal of the system is to provide the user with information on what to do in his or her current location and where to go next. Therefore, the goal is more general than it is for similar recommendation systems focused instead on exclusively recommending restaurants [46][62][79] or tourist attractions [43].



29.4.2012, 17:13
Gas distribution station
karol
not rated yet

Figure 27. *Examples of photos and GPS trajectories from a Mopsi user's collection.*

Recommendations can be based on current user location [70], and recommended items can be events nearby [51]. We use location history as a relevance criterion, as does CityVoyager [76] and the system described in [96], which considers three factors in the recommendation process: user location history, similarity between users in terms of the location history and prediction of individual interests, and user preferences. We aim to predict user activities and detect patterns in user behavior by analyzing their trajectories as described in Chapter 3. The recommendation system that recommends locations and activities based on collection of trajectories is described in [93]. The system extracts points of interest from trajectories. However, extraction is carried out based on an analysis of descriptions of activities added by users to the recorded trajectories and the database of points of interest is built manually.

Predictions on user activity are used in many recommendation systems [10]. Recommendation systems use collaborative filtering methods in association with additional field-specific methods [78][88]. In addition to collaborative filtering, distance to the recommended item also plays an important role [49]. In [25][44][69] and [91], the systems focus on user experience by considering user preferences, time, location and similarity between users to predict the user's future behavior. Similar to our system, the data from a user's social network and transportation mode is inferred so that user input is minimal and the user does not provide any explicit information used in the recommendation process.

4.2 ASPECTS OF RELEVANCE

Four aspects of relevance in sharing location-based media have been identified in [30]. Content of data, location, time, the user and his or her social network are shown in Fig. 28. The picture taken by user Pasi and shows skis and winter landscape, thus it relates to wintertime. The date when the photo was captured is important to distinguish this photo among others as it shows that it was still possible to ski in April. The location shows where it was possible to ski. The identity of the user is important as strangers may not benefit from information about available skiing tracks, but friends of the user who share the same hobby will find the information useful.

According to [30] the most important aspects of relevance are: location, content and time. In our recommendation system, location is considered the most important aspect of relevance. We assume that only items that are nearby or easily accessible from the location of the user are relevant.

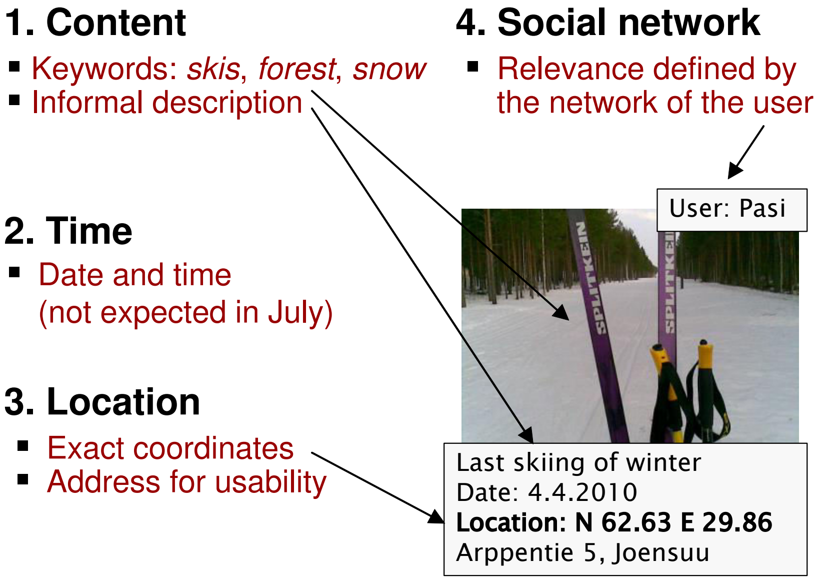


Figure 28. *Examples of four aspects of relevance on a geotagged photo.*

Users traditionally define content by keywords and tags [52]. However, keywords are less user-friendly than freeform text, particularly in social media [30]. In our recommendation system, we therefore only use freeform descriptions of photos in order to give greater freedom to users and minimalize their efforts while taking a picture. Nevertheless, extending the recommendation system would be possible by analyzing the content of the photos based on color, texture and shapes and tagging each photo automatically with predefined keywords.

Time can add relevance in several ways. For example, it is important to only suggest places or events that are available. Concerts or sport events are of course only relevant within a limited time period. Some activities, such as skiing, are only available in a certain season. Time is also important when considering the age of the photo. Newer information is more likely to be still valid than older [30].

4.3 CONTEXT-AWARE RECOMMENDATION SYSTEM

The recommendation system in [III] is designed to recommend what to do in a current location and where to go next. It uses the four aspects of relevance as recommendation context and, unlike the search algorithm implemented in Mopsi [31], we do not use any explicit user input to find relevant items. Our recommendation system is an example of a hybrid recommendation system as it uses both collaborative and content based filtering because collaborative filtering alone is not enough for providing relevant recommendation [27][35]. Collaborative filtering methods analyze information about users' behaviors and preferences and the likelihood of users predicting what users may like. We employ user-based collaborative filtering that relies on user similarity [40]. Content-based recommendation systems recommend items that are similar to those that users liked [16].

A typical case scenario is that user asks for a recommendation in a certain location via mobile phone. We exploit location awareness and the ubiquity of mobile devices [66]. The recommendation is also implemented on the website but mobile access is the most natural environment where the system can be beneficial to real life.

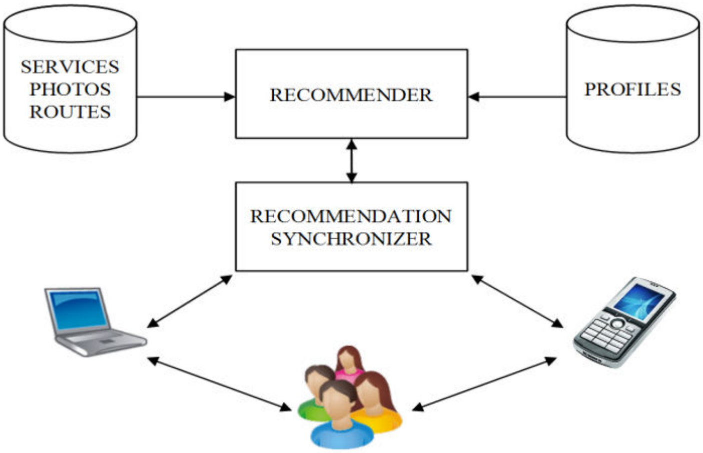
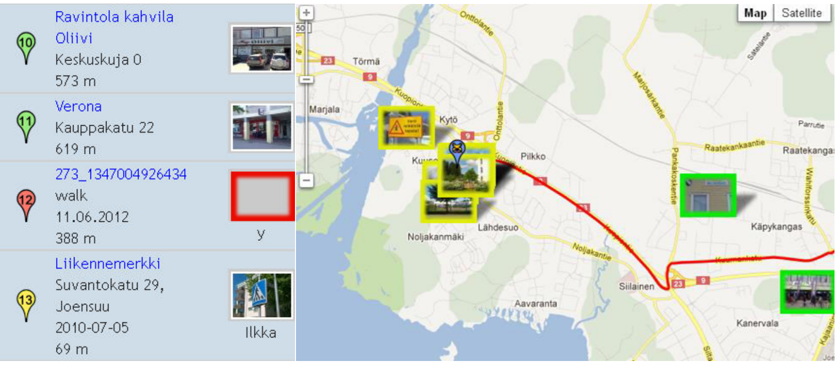


Figure 29. *Architecture of the recommendation system.*



10 Ravintola kahvila Oliivi
Keskuskuja 0
573 m

11 Verona
Kauppakatu 22
619 m

12 273_1347004926434
walk
11.06.2012
388 m
y

13 Liikennemerkki
Suvantokatu 29, Joensuu
2010-07-05
69 m
Ilkka

Figure 30. *Recommendation system results on website. Upper screenshot shows result list. Lower screenshot shows results on map.*

The recommendation system uses three databases of recommended items: services, photos and GPS trajectories as shown in Fig. 29 and the recommendation results presented in Fig. 30 contain items of each of the types mentioned above.

Recommendations are calculated on a server for every user of the Mopsi system in advance. When a user's location changes, this enables the recommendation to be provided in real-time. Changes in location are detected by the mobile application, or when the user drags the location pointer on the map in a web-based system. However, location change is not the only criterion used in making a decision on whether or not to recalculate recommendations. The decision to recalculate is carried out by a recommendation synchronizer that checks whether the results need to be recalculated or that the available recommendation is still relevant. The recommendation results are recalculated whenever a user's position moves more than 500 meters, or when the recommendation results become older than one day.

For each user of the Mopsi system, we create a profile that contains data about behavior such as location history and previous interactions with the system. For each user we store the location and keyword of every search made. Based on this data, we compile a list of the most popular keywords within their counts. In addition, we record information on what service or photo a user rated. Based on that we create a list of liked and disliked keywords. Information about viewed services, photos and trajectories are also stored.

The recommendation algorithm has three major steps: first, the items are sorted by the distance to the location of the recommendation request and items further than 2.2 km are filtered out. Second, the remaining items are scored. These first two steps are carried out for any type of item. Third, the items of different types are merged and ranked according to their scores. The first 20 items form the final result of the recommendation.

4.4 SCORING SERVICES

The recommendation algorithm has three input parameters: the profile of the user who requested recommendation, the location of the user and the time of the recommendation request.

Services are scored using history, distance and rating criteria as shown in Fig. 32. The recommendation score of each service is calculated using the following formula:

$$S_{Service} = N_H + 2N_L + N_R + 1 \tag{12}$$

where N_H is the normalized score for search history, N_L is the normalized score for location, and N_R is the normalized score for rating. For services, a constant of one point is added in order to promote services for recommendation, because they are assumed to originate from a trusted source and therefore be more relevant than photos from a user's collection. For example, cafeterias in the service database can be assumed to exist in real, but cafeterias in user collections may no longer exist if the photo was taken a long time ago.

The final score is the weighted sum of the normalized scores. In the current version of our system, the weights of the normalized score are set based on experimental results. All weights are 1, except for the location weight, which is set to 2 in order to emphasize the importance of the location.

The history score of service x is based on keywords in the search history of all users (S_G) and in the search history of the individual user asking for recommendations (S_U). Both the search history of all users and of the individual user consists of three sub-scores, given as follows:

$$\begin{aligned} S_G(x) &= S_N(x) + S_R(x) + S_F(x) \\ S_U(x) &= S_N(x) + S_S(x) + S_F(x) \end{aligned} \tag{13}$$

where S_N is score based on the searches performed near the location of the recommendation request (2.2km), S_S is based on recent searches within the last week, and S_F is based on the frequency of service keywords in search history. The details on how the frequency score is calculated is shown in Fig. 31.

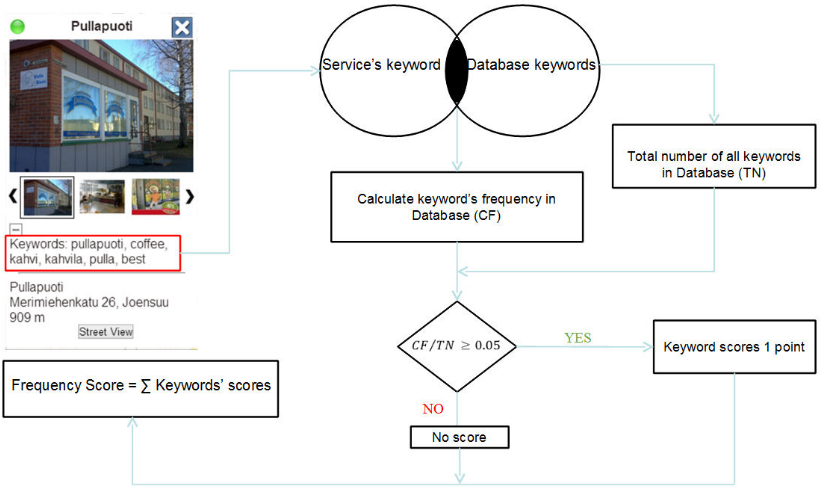

Pullapuoti
Keywords: pullapuoti, coffee, kahvi, kahvila, pulla, best
Pullapuoti
Merimiehenkatu 26, Joensuu
909 m
Street View
Frequency Score = ∑ Keywords' scores
Service's keyword
Database keywords
Total number of all keywords in Database (TN)
Calculate keyword's frequency in Database (CF)
$CF/TN \geq 0.05$
YES
Keyword scores 1 point
NO
No score

***Figure 31.** Frequency score calculation for a service.*

The total score for search history (S_H) is calculated as follows:

$$S_H(x) = S_G(x) + S_U(x) \tag{14}$$

Location score is based on the distance between item x in a set of the scored items X and user location L. The location score is calculated based on the following formula:

$$S_L(x, L, X) = \max(d(x, L), x \in X) - d(x, L) \tag{15}$$

where d is a haversine distance function.

The services have been rated by users in the scale from 1 to 5. The rating score is calculated as the average value of the n ratings:

$$S_R(x) = \frac{\sum_{i=1}^{n} R_i}{n} \tag{16}$$

Before calculating the total score, these scores are first normalized to the scale [0..1] as follows:

$$N(s) = \frac{s - min(S)}{max(S) - \min(S)} \tag{17}$$

where $s \in S$ is the raw score of an item x in a single criterion (history, location, rating or time), and S is a set of scores of all scored items using the respective criterion.

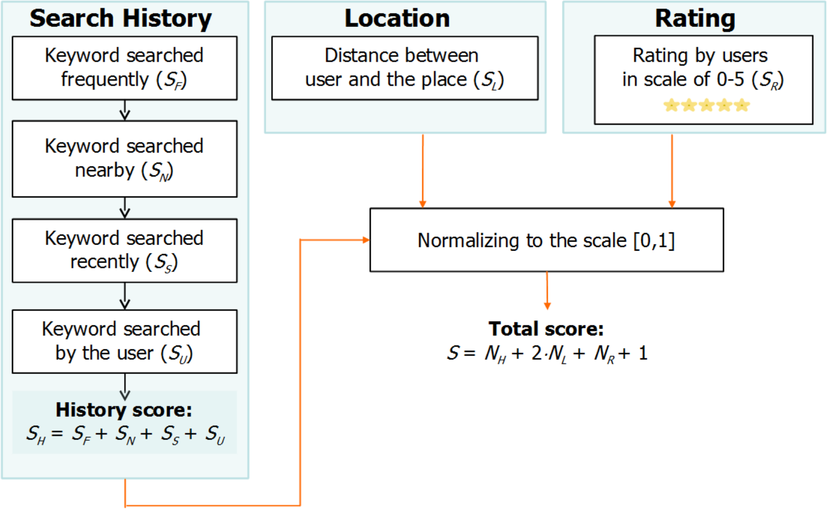

Search History
Keyword searched frequently (S_F)
Keyword searched nearby (S_N)
Keyword searched recently (S_S)
Keyword searched by the user (S_U)
History score:
$S_H = S_F + S_N + S_S + S_U$
Location
Distance between user and the place (S_L)
Rating
Rating by users in scale of 0-5 (S_R)
Normalizing to the scale [0,1]
Total score:
$S = N_H + 2{\cdot}N_L + N_R + 1$

***Figure* 32.** *Service scoring.*

4.5 SCORING PHOTOS

Photos are scored using history, distance, rating and time criteria. The first three criteria are the same as for services. However, photos do not have keywords assigned to them, therefore, we take words from the freeform description of the photo and use them as a set of keywords describing the photo. Instead of rating them from 1 to 5, photos are scored using a thumbs up (+1) and thumbs down (-1) system. The rating score is calculated by summing up these ratings:

$$S_R(x) = C_+(x) - C_-(x) \tag{18}$$

where C_+ is the number of all thumbs up and C_- is the number of all thumbs down. The time score is used, because the relevance of a photo decreases with time and depends on the time of the year that it was taken. For instance, for cross-country skiing, tracks are less relevant in summer. Recent photos in the collection are considered more relevant and get a higher up-to-date score S_A. Photos that have been taken in the season in which the recommendation has been requested get a higher season score S_Y. The time score (S_T) is for each photo is calculated as follows:

$$S_T(x) = S_A(x) + S_Y(x) \qquad (19)$$

As a final step in photo scoring, we group the photos into location clusters. We process all the scored photos iteratively. If a photo does not yet belong to any cluster then we check to see if there is any photo belonging to a cluster within a 20 m radius. In such a case, the photo is added to the cluster, so expanding its area. Otherwise, the photo creates a new cluster. From each cluster, we select at most the two highest scored photos. This helps us to avoid recommending redundant photos from the same point of interest.

4.6 SCORING TRAJECTORIES

GPS trajectories are scored using location, time, and attractiveness criteria. We select trajectories based on the proximity of their starting points to the location of the recommendation request. We consider only trajectories longer than 1 km, that have end point outside 1 km radius from the starting point. The scores for location and time are calculated exactly the same way as for services and photos. The distance is calculated from the starting point of the trajectory.

Trajectory attractiveness is calculated based on how many photos, services and other trajectories' starting points are in proximity of the end point of the trajectory. The number of photos along the trajectory (popularity) also increases its attractiveness. The total trajectory attractiveness score S_A of trajectory x is calculated using the following formula:

$$S_A(x) = S_D(x) + S_P(x) \qquad (20)$$

where S_D is destination attractiveness score and S_P is popularity of the trajectory. As with the photos, clustering is applied to the end points of trajectories. From each location cluster, we select a trajectory with the highest score so that we recommend only one trajectory for the same destination.

Fig. 33 shows an example of the best service, photo and trajectory in the recommendation results.

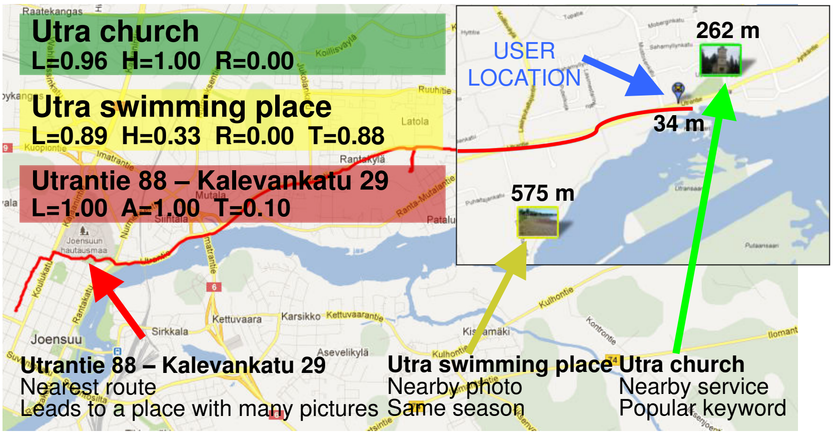


Figure 33. *Recommendation scores for three best items: a service, a photo and a trajectory.*

4.7 EXPERIMENTS

An example of recommendation results is shown in Fig. 34. User satisfaction with the recommended items is an important part of evaluating the recommendation system [3]. A common approach to evaluate recommendation systems is by giving out questionnaires such as those described in [63]. However, such method is also subjective; dependent on the user. To evaluate our recommendation system, we have chosen town of Joensuu, Finland, as it is the location with the most extensive collection of recommended items and the most users in the area.

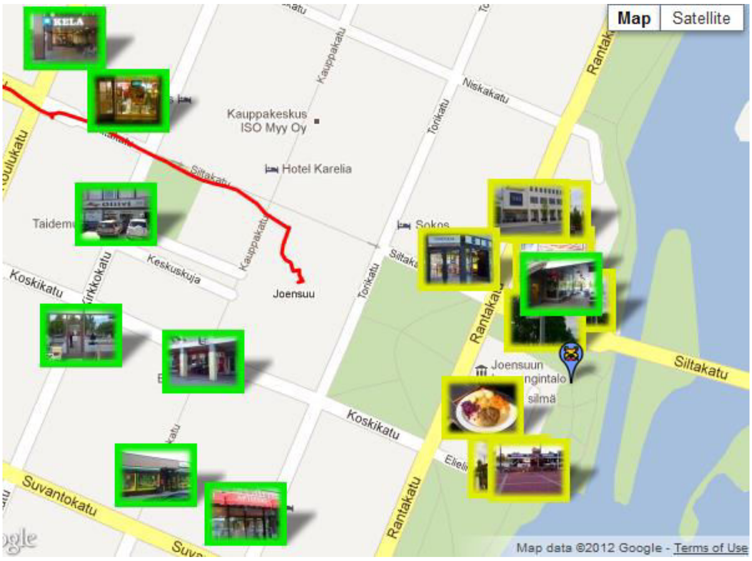
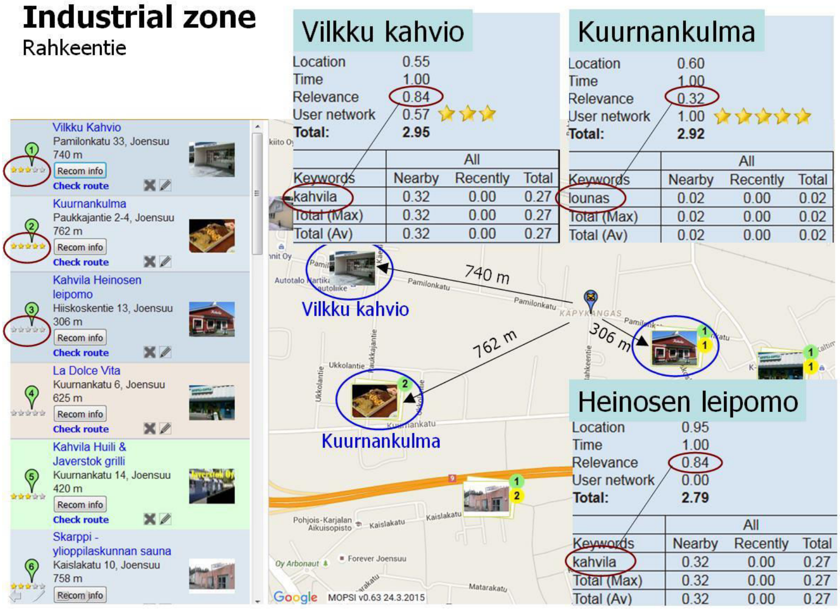


Figure 34. *Recommendation in the center of Joensuu.*

We selected several locations around town including the city center, residential district (Rantakylä), industrial area (Käpykangas) and nature area (Utra). We collected feedback in form of interviews on how relevant recommendations are for users. Each participant was shown recommendations generated by the system for his or her account and for anonymous account in each of the locations. Next we perform brief qualitative analysis of the results by showing few selected examples. Quantitative analysis was not performed due to lack of large-scale ground truth data.

The users where asked to evaluate relevance and usefulness of the results. The recommendation system in the center of Joensuu always suggests a number of cafeterias, restaurants and bars from the services database. However, the services photographed by users, but not yet added to the services database, are also recommended.

The experiments show that all factors have an impact on the selection of recommended items. For example, in the residential district, many restaurants and bars located in center are recommended, but they are selected based on rating and search history, rather than by distance. We checked that the system usually chooses relevant photos of shops, sports grounds and other services, whereas photos of streets, houses and people are skipped over, even despite their nearby location. Nevertheless, in areas where user collection is sparser, distance affects relevance scores more significantly.



Industrial zone
Rahkeentie

Vilkku Kahvio
Pamilonkatu 33, Joensuu
740 m
Recom info
Check route

Kuurnankulma
Paukkajantie 2-4, Joensuu
762 m
Recom info
Check route

Kahvila Heinosen leipomo
Hiiskoskentie 13, Joensuu
306 m
Recom info
Check route

La Dolce Vita
Kuurnankatu 6, Joensuu
625 m
Recom info
Check route

Kahvila Huili & Javerstok grilli
Kuurnankatu 14, Joensuu
420 m
Recom info
Check route

Skarppi - ylioppilaskunnan sauna
Kaislakatu 10, Joensuu
758 m
Recom info

Vilkku kahvio

Location	0.55
Time	1.00
Relevance	0.84
User network	0.57 ★★★
Total:	**2.95**

	All		
Keywords	Nearby	Recently	Total
kahvila	0.32	0.00	0.27
Total (Max)	0.32	0.00	0.27
Total (Av)	0.32	0.00	0.27

Kuurnankulma

Location	0.60
Time	1.00
Relevance	0.32
User network	1.00 ★★★★★
Total:	**2.92**

	All		
Keywords	Nearby	Recently	Total
lounas	0.02	0.00	0.02
Total (Max)	0.02	0.00	0.02
Total (Av)	0.02	0.00	0.02

Heinosen leipomo

Location	0.95
Time	1.00
Relevance	0.84
User network	0.00
Total:	**2.79**

	All		
Keywords	Nearby	Recently	Total
kahvila	0.32	0.00	0.27
Total (Max)	0.32	0.00	0.27
Total (Av)	0.32	0.00	0.27

Figure 35. Effects of relevance, history and user network.

Fig. 35 demonstrates effects of relevance, history and user network on position on recommendation list. For the user in this example cafe (kahvila) is more relevant than lunch (lounas) due to his search history. Therefore, Vilkku and Heinosen Leipomo cafes have higher relevance than Kuurnankulma lunch restaurant. In addition, Heinosen Leipomo is relatively new and has no rating yet. Thus, it has zero score from the user network, and is ranked third despite being closest.

Fig. 36 demonstrates that the time score is very important for the relevance of recommended items. The recommendation system suggests visiting the island on the lake in wintertime when it is accessible by skis and is a popular place for picnicking. On the other hand, in summertime, there is no possibility of reaching the island without a boat and not many people visit it, thus the system recommends the closest attractive places on the shore of the lake.

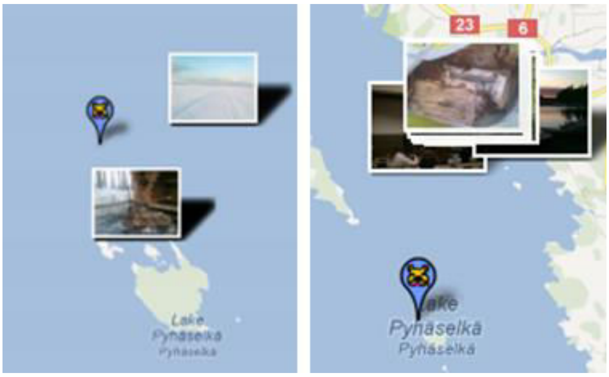


Figure 36. *Recommended photos in winter (left) and summer (right).*

In Fig. 37, we demonstrate the effect of clustering the photos. Without clustering the system recommends several pictures of the same building; while the building itself may be relevant, repeated pictures of the same building are not. After clustering, only one representative picture is recommended.

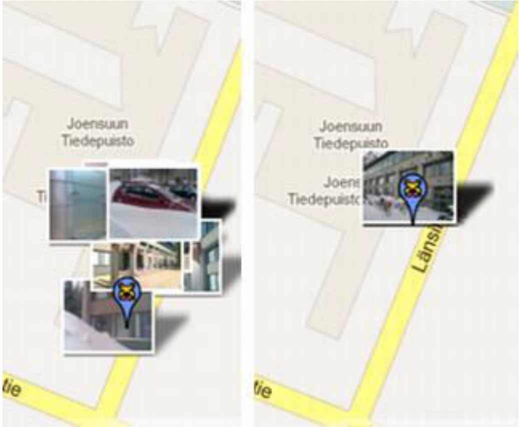


Figure 37. *Photo recommendation without location clusters (left) and with clustering applied (right).*

According to our qualitative evaluations, the current version performs reasonably well and usually gives relevant recommendations. The proposed recommender system improves over existing systems that it is not restricted to recommend one specific type of services such as restaurants and tourist attractions from database, but utilizes content generated by users. Nevertheless, it is possible to enhance its performance by measuring user similarity to provide personalized recommendations, as will be described in the following chapter.

5 User Similarity

5.1 MOTIVATION

So far, the network aspect has only been used to a limited extent, mainly because there is no social network established in the Mopsi system. To further develop the recommendation system in this direction, we study how to calculate user similarity.

Knowledge from social networking was utilized in [7] by asking local experts' opinions. Such an approach can be used to increase value in the rating of points of interest by utilizing the experience of users who know the most about them. The user network can also help to solve 'cold start' problems for newly registered users or users with a limited location history. Profiles of friends of such users help to personalize recommendation results as soon as a user joins the network [89]. Collaborative recommendation systems commonly establish similarities between users that are based on common items the users have rated [1]. The similarity of users is applied to recommending events and friends in [26].

It is, however, unclear what type of network can best be used to discover similarity between users. We consider three possible data sources for calculating user similarity: friendships in Facebook, pages liked in Facebook and location history in Mopsi. For the two first data sources, we perform a qualitative experiment with nine Mopsi users. For location history, we study the similarity of their sparse location histograms. According to [38], people most value the things they have in common, followed by the place where they are active and finally knowing the same people. Therefore, we hypothesize that page 'likes' on Facebook can be considered a better calculation of user similarity than location history, as 'likes' show users' own evaluation of the similarity between them. However, the similarity of location history provides additional information for the personalization of the recommendation.

We aim to calculate similarity between users based on a limited amount of location data. There are attempts to analyze complete GPS trajectories and deduce user similarity once the results have been analyzed [22]. The approach used in [90] is to detect stops that are considered to be important places since users stayed there for a longer time. Subsequently, the similarity of trajectories is then measured based on their longest common subsequence with higher importance given to longer patterns. Similarly, in [54] the longest common subsequence is applied by partitioning the complete GPS trajectory and detecting turning points. The similarity score is a combination of the geographic and semantic similarity of the trajectories. The semantic meaning of GPS trajectories is also used in [12] to find the similarity of daily schedules of people. A personalized search for similar trajectories is carried out in [83], by taking user preferences into account where parts of the 'query trajectory' are considered to be more important.

However, complete trajectories are not always available. In such cases, user similarity must be measured based on sparse location data such as visits, favorite places or check-ins. In [53], user data is hierarchically clustered into geographic regions. A graph is constructed from the clustered locations so that a node is a region a user has visited and edges between the nodes represent the order of visits to the regions represented by these nodes. The method does not require a complete trajectory, but it still relies on the order of the visits to the locations.

We use single location points that originate from users' collections and represent locations of photos and the start and end points of their trajectories. In future, this location history dataset can be extended with end points where user stayed longer than a certain threshold [94]. The end points can be detected by the segmentation algorithm that was described in [II].

In our method, we compare histograms. Histograms represent points of interest in the area and each user's activity is assigned to the nearest point of interest. We consider several measures based on normalized frequency vectors: L_1, L_2, L_∞, ChiSquared, Bhattacharyya and Kullback-Leibler divergence. We compare the performance of fuzzy and crisp histograms.

5.2 EFFECTIVENESS OF USER NETWORK

There are various types of user networks on social web sites [45]. We evaluated the possible types of networks that can be used by our recommendation system from the following perspectives: social network versus information sharing network, buddy network versus stranger network, selected friends network versus automatic ad hoc network, and online network versus offline network.

Currently the networks with the highest number of users are the social networks of Facebook, Twitter, Google+ and Instagram. In these social networks, users explicitly specify with whom they share their data. Users are generally more interested in data coming from their friends. Nevertheless, for the purposes of our recommendation system, relevance is far more important than social aspect. In a location-aware system like ours, users look for information about their current location. Therefore, information originating from a stranger who resides in the area or visits it is more likely to be relevant than information from a friend who has never visited the location.

An aspect worth considering for any social network is how well the connected people actually know each other (in real life). The small world phenomenon [84] says that any person in the world can be reached in six steps. As shown in [9] the quality of the links within the network, it is important, while connectivity is of less importance because even a small amount of randomness in the network can trigger small world behavior.

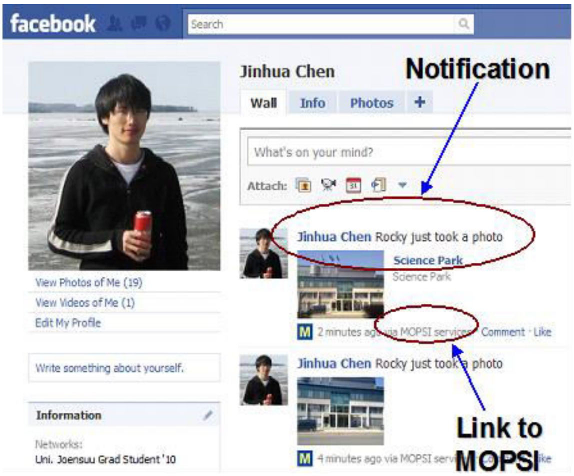
The Facebook network, despite the term used on Facebook, is not really a friend network, but more a buddy network. Because of social pressure, Facebook users often send friend requests to many other users they may not know. Nevertheless, having hundreds of Facebook friends does not imply that the person has hundreds of real friends. It means, though, that the person has hundreds of acquaintances while the number of friends most likely does not exceed 10. As the people connected on Facebook know each other, the small word phenomenon applies to it. The strength of the connection in the network is more important than the connection itself for the distribution of information. Sharing information with a large number of weakly connected people is likely to be less effective than sharing it with a smaller number of strongly connected users.

Nevertheless, unlike in social networks, people who have never before met might be connected by common interests. For example, couchsurfing is a network of people who offer places to sleep for travelers without financial compensation [15]. However, in such networks, trust between users needs to be firmly established in order for the system to work.

There are ways to establish networks and link users automatically. Users can be connected, for example, by their behavior [36], or, as in Mopsi system, by location. Such automatically discovered connections can be useful for giving recommendations. Similar users can be recommended to each other as in [6].

Another ad hoc network, described in [85], uses face analysis techniques to identify people in photos. Based on face recognition results, a social network is created by linking people tagged in photos to each other. Analyzing photos also gives the possibility of connecting, for example, people with the same hobbies if a more detailed content analysis is performed.

Another approach is to combine a location-based service and a social network from two independent systems, as shown in [72]. Following up this approach, we connected Mopsi with Facebook so that users can share their collected photos and trajectories by using Mopsi on Facebook. An example of Mopsi activity posted on Facebook is shown in Fig. 38.



Jinhua Chen

Jinhua Chen Rocky just took a photo

Science Park

Science Park

2 minutes ago via MOPSI services

Jinhua Chen Rocky just took a photo

4 minutes ago via MOPSI services

Uni. Joensuu Grad Student '10

Figure 38. Facebook status update when uploading a Mopsi photo.

Depending on the nature of the network, online or offline, users behave differently. The Mopsi system has no online forum, thus it has an offline nature, even though data collection is often done online. Personality also affects how people use social networks. Extraverted personalities are more likely to engage in social activities but according to [67], personality has a much smaller effect on how they use Facebook than previously expected. For example, a social person is likely to join more groups but that does not have much bearing on the size of that network, or how extensively the communicative functions are used. People with a higher sensitivity to threat use more textual expression and less photo sharing because it is more controllable due to its offline nature [67]. Another study showed that the identity people present in their social network can differ a great deal from their real personalities. It was observed that the images people gave were more real in off-line chatting environments than they were in offline social networks [92].

Many people are concerned about their privacy in social networks. People often do not wish to reveal their current location or their identity. There are methods to prevent the system combining a user's identity and location [75]. The privacy issue, if not adequately addressed, may weaken information sharing within social networks.

5.3 PROFILE SIMILARITY

To check the similarity of users in the Mopsi system, we selected nine volunteers (shown in Table 4). All of the selected users live in Joensuu, use both Mopsi and Facebook and know each other to some extent. However, not all of them are linked on Facebook. We asked them to evaluate their level of acquaintance with the other volunteers on a scale of 1 to 8 by answering two questions: how similar is the person to you, and, how useful to you are his or her Mopsi photos. The first question was strictly meant to measure similarity. With the latter question, we wanted to find out if a user recommends useful and interesting places to visit via his or her Mopsi photos. The rankings prepared by the volunteers are shown in Tables 5 and 6. The pink background of certain cells is used to indicate that the users are not linked in Facebook. As expected, if one considers the other similar, then they are also connected in Facebook. Similarity seems to correlate with connection in the social network.

Analysis of the questionnaire results shows that the similarity ranking is subjective. The average values show that three users – Radu, Pasi and Andrei – have the most connections with all the other users. Specifically, Radu is the most similar to five users and ranks 2nd or 3rd for the remainder. A further analysis of Facebook activity data shows that the more photos and status updates of a user is liked and commented on, then the more similar the user is considered to the users who liked and commented. The two rankings made by volunteers correlate with each other. Nevertheless, there are certain differences. For example, Pasi is the highest ranked user in terms of the usefulness of his photos. Similarly, Julinka’s photos are considered useful. The data of the two users is considered useful, because both of them frequently publish travel photos. Even though we asked how useful users *expect* the data of their friends to be, expectation may not match reality. Some low rankings might be biased towards low publication activity rather than the actual usefulness of these photos.

Table 4. *Summary of volunteer data in Mopsi and Facebook.*

	Mopsi			**Facebook**	
	photos	places	visits	friends	pages
Andrei	676	96	676	463	285
Julinka	3850	122	2116	229	154
Mikko	190	84	292	55	14
Oili	6467	164	1261	298	63
Pasi	9716	208	3847	88	67
Radu	1417	122	912	298	19
Rezaei	716	85	587	193	16
Chait	63	22	53	580	195
Jukka	991	126	682	142	120

Table 5. *User similarity based on their own rankings.*

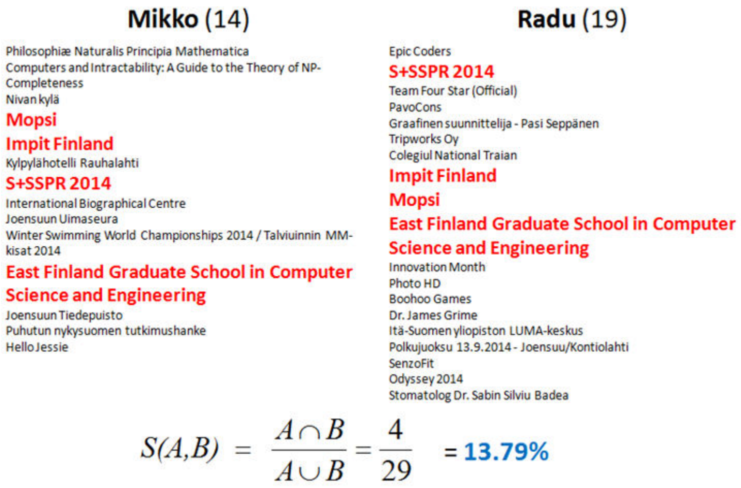
	Andrei	Julinka	Mikko	Oili	Pasi	Radu	Rezaei	Chait	Jukka
Andrei	-	7	8	4	2	1	3	6	5
Julinka	2	-	4	3	6	1	5	7	8
Mikko	7	8	-	5	1	2	4	6	3
Oili	3	5	7	-	2	1	4	8	6
Pasi	3	8	5	4	-	2	6	7	1
Radu	1	8	4	5	2	-	3	7	6
Rezaei	4	7	2	6	1	3	-	8	5
Chait	2	8	4	7	5	1	3	-	6
Jukka	2	7	5	4	3	1	8	6	-
Avg.	**3.0**	**7.3**	**4.9**	**4.8**	**2.8**	**1.5**	**4.5**	**6.9**	**5.0**

Table 6. *Ranking of the usefulness of photos.*

	Andrei	Julinka	Mikko	Oili	Pasi	Radu	Rezaei	Chait	Jukka
Andrei	-	5	8	4	1	2	6	7	3
Julinka	2	-	6	3	4	1	5	7	8
Mikko	4	1	-	8	2	6	7	5	3
Oili	4	5	7	-	1	2	6	8	3
Pasi	2	7	1	4	-	5	8	6	3
Radu	2	5	7	4	1	-	6	8	3
Rezaei	6	2	7	3	1	5	-	8	4
Chait	3	7	8	4	2	1	6	-	5
Jukka	3	6	5	4	1	2	8	7	-
Avg.	**3.3**	**4.8**	**6.1**	**4.2**	**1.6**	**3.0**	**6.5**	**7.0**	**4.0**

We measured the similarity of users by checking the number of the same Facebook pages they liked. For example, Mikko and Radu like four out of 29 of the same pages that either one or both like on Facebook. We define their similarity by the Jaccard similarity coefficient, i.e. the number of matches divided by the total number of pages, as shown in Fig. 39. Liking exactly the same page in the larger scale is not likely to happen. For example, a user who likes Hesburger, a local fast food restaurant brand, and another user who likes McDonald's are likely to be similar in the sense that they both like fast food. However, they are not similar based on the same page likes. Therefore, we consider counting the matches of page categories on Facebook instead of pages themselves and applying the same Jaccard similarity coefficient as shown in Fig. 40. However, category based matches are unlikely to be useful for recommendation purposes as the categories tend to be very general.

The similarity coefficient values are the smallest among people who are not linked on Facebook. Liking a page also correlates reasonably well with the user similarity values (Table 5) but the correlation with usefulness values (Table 6) is about three times smaller. Therefore, even if user similarity could be estimated by their user profiles in Facebook, using it for a location-aware recommendation would still be questionable.



Mikko (14)

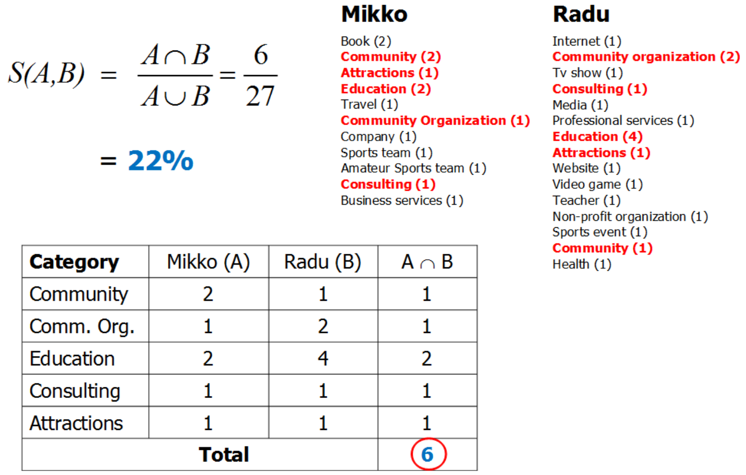
Philosophiæ Naturalis Principia Mathematica
Computers and Intractability: A Guide to the Theory of NP-Completeness
Nivan kylä
Mopsi
Impit Finland
Kylpylähotelli Rauhalahti
S+SSPR 2014
International Biographical Centre
Joensuun Uimaseura
Winter Swimming World Championships 2014 / Talviuinnin MM-kisat 2014
East Finland Graduate School in Computer Science and Engineering
Joensuun Tiedepuisto
Puhutun nykysuomen tutkimushanke
Hello Jessie

Radu (19)

Epic Coders
S+SSPR 2014
Team Four Star (Official)
PavoCons
Graafinen suunnittelija - Pasi Seppänen
Tripworks Oy
Colegiul National Traian
Impit Finland
Mopsi
East Finland Graduate School in Computer Science and Engineering
Innovation Month
Photo HD
Boohoo Games
Dr. James Grime
Itä-Suomen yliopiston LUMA-keskus
Polkujuoksu 13.9.2014 - Joensuu/Kontiolahti
SenzoFit
Odyssey 2014
Stomatolog Dr. Sabin Silviu Badea

$$S(A,B) = \frac{A \cap B}{A \cup B} = \frac{4}{29} = 13.79\%$$

Figure 39. *User similarity calculation based on page likes on Facebook.*



$$S(A,B) = \frac{A \cap B}{A \cup B} = \frac{6}{27}$$

$$= 22\%$$

Mikko

Book (2)
Community (2)
Attractions (1)
Education (2)
Travel (1)
Community Organization (1)
Company (1)
Sports team (1)
Amateur Sports team (1)
Consulting (1)
Business services (1)

Radu

Internet (1)
Community organization (2)
Tv show (1)
Consulting (1)
Media (1)
Professional services (1)
Education (4)
Attractions (1)
Website (1)
Video game (1)
Teacher (1)
Non-profit organization (1)
Sports event (1)
Community (1)
Health (1)

Category	Mikko (A)	Radu (B)	A ∩ B
Community	2	1	1
Comm. Org.	1	2	1
Education	2	4	2
Consulting	1	1	1
Attractions	1	1	1
Total			6

Figure 40. *User similarity based on category likes on Facebook.*

User similarity correlates with all the analyzed features. Nevertheless, the correlation is not very strong. In most cases, users ranked people they are already connected to on Facebook as the most similar. There is also a strong correlation with the pages liked on Facebook.

5.4 LOCATION HISTORY SIMILARITY

For studying similarity based on location history, we selected 293 points of interest from the Mopsi database. These points of interest include hotels, restaurant, cafeterias, holiday resorts, shops, recreational places and many others that Mopsi users consider relevant. The locations of these points of interest are used as histogram bins, which we denote as places. For the purpose of this study, we only consider the Joensuu sub region, as shown in Fig. 41. However, the area is diverse as it covers downtown Joensuu and the suburbs, bordering municipalities and rural areas. As seen in Fig. 41, the places of interest are dense in the downtown area, but sparse in rural areas. For example, there are places that are approximately 20 km away from another places, but in downtown Joensuu there are about 75 points of interest located in 1.5 sq. km around the market square alone.

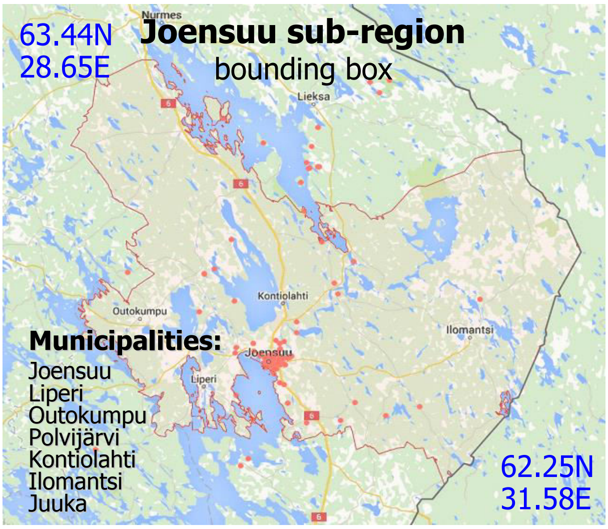


__Figure 41.__ Distribution of places of interest in the Joensuu sub region.

User similarities are calculated based on the locations of where the users collected data, namely photo points and start and end points of GPS trajectories. We refer to these points as activity points.

We show the Bhattacharyya distance measure for users from Chapter 5, Section 3 in Table 7.

***Table 7.** Bhattacharyya distance similarities.*

	Andrei	Julinka	Mikko	Oili	Pasi	Radu	rezaei	Chait	Jukka
Andrei	-	0.33	0.32	0.34	0.54	0.5	0.51	0.38	0.45
Julinka	0.33	-	0.29	0.45	0.52	0.4	0.4	0.46	0.35
Mikko	0.32	0.29	-	0.27	0.53	0.59	0.38	0.3	0.37
Oili	0.34	0.45	0.27	-	0.46	0.37	0.51	0.6	0.3
Pasi	0.54	0.52	0.53	0.46	-	0.68	0.68	0.52	0.54
Radu	0.5	0.4	0.59	0.37	0.68	-	0.58	0.45	0.65
rezaei	0.51	0.4	0.38	0.51	0.68	0.58	-	0.53	0.56
Chait	0.38	0.46	0.3	0.6	0.52	0.45	0.53	-	0.42
Jukka	0.45	0.35	0.37	0.3	0.54	0.65	0.56	0.42	-

For further experiments, we collected the location of the three users from the years 2011–2014. The most active users in the area were selected: Andrei (A), Pasi (P) and Radu (R). We call these three users the APR trio in the following. A summary of the location data divided into years is shown in Table 8. We show the distribution points for the trio in Fig. 42. The APR trio has almost 6,000 activity points in total. The most popular points, together with the corresponding visit frequencies, are listed in Table 9.

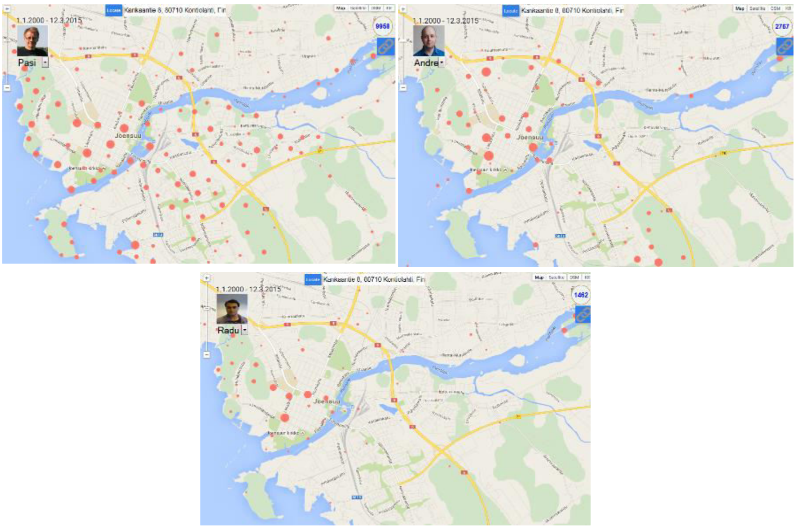


***Figure 42.** Users' activity points in downtown Joensuu.*

Table 8. Three test users and the summary of their activity points.

	2011	2012	2013	2014
Andrei	206	757	432	329
Pasi	1263	545	636	751
Radu	37	292	324	259

We have twelve subsets in total. We created the artificial users by dividing the data of each user by year. The subsets are denoted as: A11, A12, A13, A14, P11, P12, P13, P14, R11, R12, R13, R14 (A11 is the data of user Andrei from year 2011, P12 is the data of user Pasi from year 2012, and so forth). During the testing of the similarity calculation methods, we expect that user A11 will be the same as user A11 and similar to users A12, A13 and A14. We observed that, in our data, the highest frequency is in the place located nearby users' individual homes. However, both Andrei and Radu moved in 2014, which resulted in changes to the corresponding histograms. All users have the same workplace (the Science Park).

Table 9. Ten most frequent entries in the histograms.

	Andrei				Pasi				Radu				
	11	12	13	14	11	12	13	14	11	12	13	14	
Niinivaara otto3	20	0	29	150	47	7	6	8	1	2	2	3	275
Salomökki	13	11	87	36	64	17	16	7	0	1	12	2	266
keskusta 1	0	0	0	0	51	54	54	69	0	1	0	0	229
Skarppi – sauna	12	107	87	0	0	0	1	2	0	0	2	0	211
Noljakan kirkko	1	0	0	1	34	9	20	11	0	0	52	54	182
Lounasravintola Puisto	6	29	10	3	6	2	1	3	7	54	35	16	172
Joensuu Areena	7	92	6	0	18	4	7	15	0	13	4	2	168
Science Park	22	6	5	4	36	9	6	18	1	12	11	21	151
Kiesa	0	3	4	0	0	0	0	1	12	82	41	7	150
Oskolan lomamökit	0	0	0	0	73	6	48	13	0	0	0	0	140

To calculate the similarity score of two users, we used the histograms created based on the users' activity points and places. We mapped the activity points to their nearest place. Every activity point increased the count of the corresponding histogram bin by one. To measure distance, we used the haversine distance between the two locations given in form of latitude (ϕ) and longitude (λ). The formula that calculates the haversine distance in km is defined as follows:

$$hav = 2 \cdot R \cdot arcsin(\sqrt{sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)} \qquad (22)$$

where R=6372.8km is the radius of Earth on the equator, ϕ_1 and ϕ_2 are the latitudes and λ_1 and λ_2 are the longitudes of the two points. Each user has n activity points mapped into m histogram bins $h(i)$ so that:

$$\sum_{i=1}^{m} h(i) = n \qquad (23)$$

The histograms are normalized according to the following formula:

$$p(i) = \frac{h(i)}{\sum_{j=1}^{m} h(j)} \forall i \in [1, m] \qquad (24)$$

where p(i) represents the probability that an activity point belongs to the i bin. A simple histogram construction example is shown in Fig. 43 where three users have n_1=9, n_2=7 and n_3=7 activity points that are mapped to m=8 places.

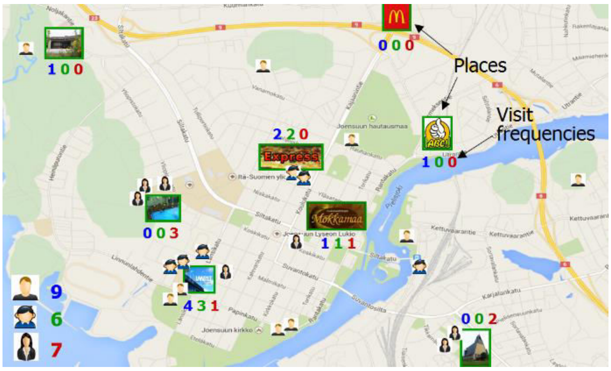
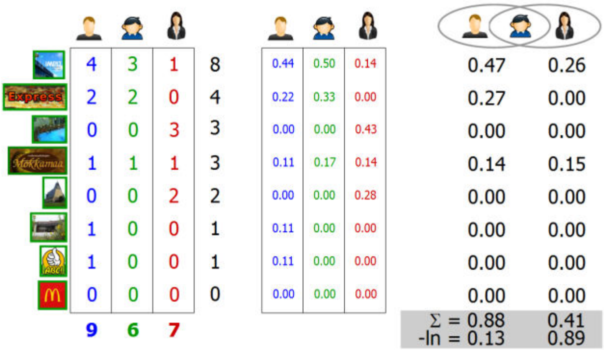


Figure 43. *Converting location history of three users to a histogram consisting of m=8 predefined places. Small icons on the map represent users' activity points, places are shown as thumbnail images and values of the bins are shown above or below each place.*

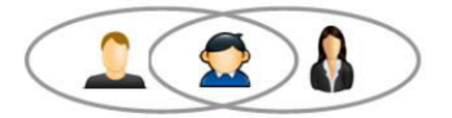
The histograms represent the probability distribution functions, and the similarity score of two users is the distance between their respective probability distribution functions. A histogram is usually a one dimensional matrix consisting of numerical values, for example, pixel intensities in a digital picture. In the example histogram, there is an explicit ordering of the bins and the values of neighboring bins highly correlate with each other. However, the bin values can be nominal or, as in our case, multivariate. Thus, there is no natural order of the bins. Nevertheless, our observations, i.e. activity points, are in a metric space and can be mapped to the histogram by simple distance calculations. By constructing such histograms, we reduced the similarity score calculation problem to a histogram comparison problem. There is extensive literature on the problem of histogram comparison and different methods exist [17][18].



4	3	1	8	0.44	0.50	0.14	0.47	0.26
2	2	0	4	0.22	0.33	0.00	0.27	0.00
0	0	3	3	0.00	0.00	0.43	0.00	0.00
1	1	1	3	0.11	0.17	0.14	0.14	0.15
0	0	2	2	0.00	0.00	0.28	0.00	0.00
1	0	0	1	0.11	0.00	0.00	0.00	0.00
1	0	0	1	0.11	0.00	0.00	0.00	0.00
0	0	0	0	0.00	0.00	0.00	0.00	0.00
9	6	7					Σ = 0.88 -ln = 0.13	0.41 0.89

Figure 44. Distance calculation between two histograms using the Bhattacharyya distance.

Table 10. Distances between the two histograms from the small example.



0.88	BHA	0.42
0.12	KLD	0.20
0.26	ChiSq	1.25
0.44	L_1	1.43
0.04	L_2	0.50
0.11	L_∞	0.43

The Bhattacharyya coefficient is one of the most commonly used distances for histogram comparison. The coefficient was originally proposed as a similarity measure between statistical populations. Fig. 44 demonstrates calculations of distance between two histograms from the small example shown in Fig. 43. Firstly, the product p_iq_i of two frequencies is calculated and its square root is then summed over all the histogram bins as shown in Table 10. Higher frequencies yield in higher product. The result is converted to a distance of range [0,1] by logarithmic scaling.

We also checked performance of other commonly used distances such as L_1, L_2, L_∞ and Chi Squared. The Kullback-Leibler distance generalizes Shannon's concept of probabilistic uncertainty – called entropy – by calculating the minimum cross entropy of two probability distributions [54]. All the distances used in the study are defined below:

$$L_1 = \sum_i |p_i - q_i| \tag{25}$$

$$L_2 = \sum_i (p_i - q_i)^2 \tag{26}$$

$$L_\infty = max|p_i - q_i|\forall i \tag{27}$$

$$d_{ChiSq} = \sum_i \frac{(p_i - q_i)^2}{p_i + q_i} \tag{28}$$

$$d_{KLD} = \sum_i \left(p_i \cdot log\frac{p_i}{q_i} + q_i \cdot log\frac{q_i}{p_i} \right) \tag{29}$$

$$d_{BHA} = \sum_i \sqrt{p_i \cdot q_i} \tag{30}$$

where p_i and q_i are the relative frequencies of the histogram bins i and the summation is carried out over all the m places.

All of the distance measurements rely on the independent calculation of the distance of each bin. In the case of sparse observations, strongly peaked histograms would become mismatched due to slight translation. The earth mover distance (EMD) aims to solve this by transforming surplus from one bin to the bins that have a deficit [68]. In the case of one-dimensional numeric data, it is straightforward to calculate by processing the histogram sequentially from left to right. Nevertheless, in multivariate case the optimal moving of the surplus becomes a more complicated problem. According to [19], the problem could be solved as a transportation problem, but faster and more efficient algorithms are needed. In [73], the peaks of the histograms were considered to be more important. The improved performance of L_1, L_2 and EMD was demonstrated for a time-series analysis by calculating the sum of the peak weights multiplied by their proximity factors.

The sparseness of the data may also cause problems in case there are too few observations in comparison to the number of bins. To solve the problem, fuzzy histograms were proposed as they were successfully applied to image processing [28]. In case of one-dimensional histograms, the observations are divided between neighboring bins. To the multivariate case, we apply the k-nearest neighbors (kNN) algorithm. For each location of activity, we find its k (nearest places). We calculate fuzzy count similarly from the fuzzy C-means algorithm [11]. The weight added to each bin is calculated based on the following formula:

$$w_i = \left(\sum_{j=1, j \neq i}^{k} \frac{d(x - h_i)}{d(x - h_j} \right)^{-1} \tag{31}$$

Afterwards, the same method as the crisp variant is used to calculate the histogram comparison.

For testing the methods we use the APR trio data. The distance (similarity score) between all possible pairs of these twelve subsets is calculated. We test if the methods can properly decide which of the subsets belongs to the same user. The threshold is used for that purpose. The expected result is that 3·4·4=48 pairs (33%) should be recognized as being the same user and 3·4·8=96 pairs (67%) should be classified as a different user, i.e. fail the test.

We study the effect of three alternative threshold techniques. Our first choice is average. We use the average of all similarity values. This is the simplest non-parametric threshold that attempts to adapt the method to the data. In our second choice (*a* *priori*) we select the threshold value that has passed 48 pairs or is as close to this number as possible. The third choice (oracle) is the threshold selected so that it provides the best accuracy for the method in question.

Table 11. *Classification error for the APR trio.*

	Threshold			Error (crisp) in %			Error (fuzzy) in %		
	Average	Apriori	Oracle	Average	Apriori	Oracle	Average	Apriori	Oracle
L_1	0.31	0.27	0.28	8	8	7	10	10	10
ChiSq	1.22	1.24	1.18	8	7	7	17	11	10
BHA	0.46	0.46	0.48	10	10	8	15	14	11
KLD	0.82	0.89	0.88	11	11	10	36	21	15
L_2	0.84	0.80	0.88	35	47	15	35	49	14
L_∞	0.79	0.72	0.87	43	43	18	38	47	21

The results in Table 11 show that L_1, ChiSquared, BHA and KLD provide good results (8%, 8%, 10%, and 11% of misclassification respectively) with the average as a threshold. The results are only slightly better if the optimal threshold was selected in the oracle method (7%, 7%, 8%, and 10% respectively). The other two methods, L_2 and L_∞, perform significantly less well (35% and 45% respectively). The a priori information does not improve results. Even when the threshold selected is optimal, the results are worse (15% and 18% respectively).

We considered fuzzy histograms with the neighborhood of fixed size k=3. The classification error increased in all the distance measurement methods, especially in KLD. Moreover, the threshold based on the average performed significantly less well; therefore, the choice of the threshold becomes critical in case of the fuzzy histograms.

The BHA and ChiSquared methods misclassified users A13 and R13. However, from the users we know that their location activity dataset was similar during 2013 as they cycled together. That caused the bins in the rural areas to have higher counts.

A further analysis of the classification errors and comparing it with knowledge we have about activity of the users shows that the performance of the methods is significantly affected by dominant values. For example, all methods recognized user A14 as a different user than A11, A12 and A13. The same happens for user R11 and R14 with all methods except KLD. Both of the real users relocated in 2014, which caused the different histograms bins to be dominant. That motivated us to run further tests on the datasets that had the top 10 bins removed. Removing the dominant bins caused weaker performance as summarized in Table 12.

Table 12. *Classification accuracy for the APR trio when 10 most popular bins have been eliminated from the calculations.*

Method	**Change**	**Observation**
L_1	8% → 24%	Loses its ability
ChiSq	8% → 13%	A11 becomes similar with P11, P13, P14
BHA	10% → 13%	A11 becomes similar with P11, P13, and P14. P11-R14 no longer match.
KLD	11% → 13%	A11 and R14 become similar, no other effects.
L_2	35% → 40%	Works even worse.
L_∞	42% → 43%	Works even worse.

Comparing the location history similarity results with the similarity of the users based on their personal views we noticed a mild correlation. Similarly, there is small correlation between the results and those expected by user usefulness of the data. Nevertheless, our study was conducted on a small group on volunteers and further research on the usefulness of the similarity measurement for recommendation purposes is needed. However, the method we designed can be successfully used to find similar users as locations of people show their interests and personal preferences.

6 Summary of Contributions

In Paper [I], we present a system that stores and visualizes GPS trajectories. The main goal of this research was to develop a system that could display multiple GPS trajectories to users in real-time. As an outcome of the research, we developed a complete, real-time web-based system for the acquisition, storage, querying, retrieval and visualization of the trajectories. The main challenge was the fast visualization of large amounts of data that was not possible with the existing online systems. We propose to reduce the quantity of data to be visualized without affecting the quality of the visualization. We achieved this goal by applying fast polygonal approximation to the GPS trajectories together with a bounding box solution for the display of trajectories and efficient database model to store the data. The developed system is able to handle the number of points that causes the existing system to stop responding or respond slowly. The system displays the most extensive collection of over 1,300 trajectories that consist of over 2.5 million points in less than five seconds.

In Paper [II], we propose the algorithm to segment the trajectory into parts with similar characteristics and detect the travel methods of each segment. GPS trajectories contain information about location and time but further statistics require analysis, which is often done using prior information such as road networks. Our method, contrary to the existing algorithms, only uses raw GPS data and does not rely on external information such as road networks, nor does it require additional information from other sources such as the accelerometer. Furthermore, no training is needed for the movement type classifier. Therefore, it can be used for classification when no training data is available.

In Paper [III], we present a recommendation system that suggests to users what to do next in their current location and where to go using the location data that has been collected by system users. Currently there is a great deal of information available on the Internet, however, it is often difficult to find information that is relevant. This is one of the reasons for applying recommendation systems that are nowadays used either explicitly or implicitly in virtually every online shop or social website. Our recommendation system finds information based on relevance to: location, time, content and social network. The system uses collaborative filtering and builds user profiles in order to personalize recommendations. We recommend items from a collection of services, geo-tagged photos and GPS trajectories that has been entirely collected by users without any supervision and not directly for purposes of recommendation but in order to interact with peers. Therefore, the system assesses the relevance of data that do not necessarily follow patterns such as photo descriptions. The database does not need to be built nor does it need to be maintained by system administrators.

In Paper [IV], we study how a user's social network can improve the results given by the recommendation system. We illustrate a method to measure users' similarities based on Facebook data. We checked how friendship and personal preferences, based on the pages liked in Facebook, correlate with user similarity. We also present a similarity measure based on sparse location history. The usefulness of the three approaches for the location-aware recommendation system is then evaluated. User similarity is best defined by friendship in Facebook, followed by the pages liked in Facebook. The location history has a smaller impact on user similarity. However, Facebook data are not always available and, in such cases, location history is the way to define user similarity. The proposed approach can be applied to every location-based service.

In Paper [V], we present a method for measuring the similarity of users based on their sparse location history that can further personalize and enhance results given by the recommendation system. In this paper, we describe the location-based similarity measurement method proposed in Paper [IV] in more detail. Previously, we only used the Bhattacharyya coefficient. Now we compare its performance with other commonly used distances such as L_1, L_2, L_∞, ChiSquared and Kullback-Leibler. We consider only sparse data about the activities of users in the area, unlike other existing methods that are mostly based on complete GPS trajectories and do not utilize geo-tagged photos. We map each activity point to the nearest location in a predefined set of points of interest. The similarity measurement problem is then reduced to a histogram comparison. We compare the six measures using both crisp and fuzzy histograms. L_1, ChiSquared, Bhattacharyya and Kullback-Leibler distances are useful for measuring user similarity based on the crisp histograms.

7 Conclusions

In this thesis, we have proposed new methods for the analysis of GPS trajectories and location-aware recommendation systems. Reduction, segmentation and movement type classifications of the trajectories have been studied in depth. The problem of finding similar users based on the sparse location history has also been addressed.

A real-time web-based system for processing and retrieving GPS trajectories was developed. The proposed system is able to handle over 1,300 trajectories consisting of over 2.5 million points in less than 5 seconds, which is a much larger amount of data than any existing systems can handle. However, there are still possibilities to improve it. For example, the number of points actually plotted on a map can be further minimized by reducing plotting overlapping trajectories. This could be achieved by clustering overlapping segments of the trajectories.

We designed a method that is able to classify GPS trajectory to five basic movement types using the second order Markov model. The method is only based on basic GPS location data without the use of an accelerometer or similar devices and it does not require prior training. The classifier can be further enhanced to detect more movement types, such as travelling by train and airplane. Another challenge is to differentiate between cross-country skiing and running.

A location-aware recommendation system was designed to recommend other users' photos and routes. Based on the recommendation, users will have suggestions about what to do in the current location and where to go next. The system selects relevant items from freeform collections of geo-tagged photos, through points of interest to GPS trajectories. The personalization of the system is carried out based on the user's previous activity. The recommendation system works in real-time by calculating the recommendations for a user in advance. However, there should be a possibility of improving the speed further by pre-computing the attractiveness score of the GPS trajectories.

The use of clustering can also be further developed. In addition to the location clusters, semantic clusters could be created by grouping together items of same type. . It would allow diversification of recommendation by suggesting items of different type. For example, we would avoid the situation when several ATMs of the same chain are recommended, as shown in Fig. 45. In future, we propose to analyze the meaning of photo descriptions. Photo content could also be analyzed and automatically assigned to predefined categories [50]. The geographic similarity of trajectories could be used to recommend the most convenient routes to reach one's destination.

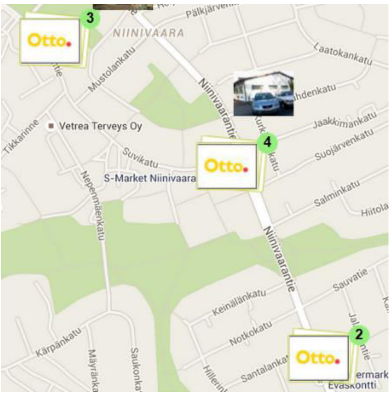


Figure 45. *Recommendation of three ATMs of the same chain.*

We studied how to find similar users in order to further personalize the recommendation results and avoid a 'cold start' problem in the recommendation system. We compared the performance of six distance measures: L_1, L_2, L_∞, ChiSquared, Bhatacharyya and Kullback-Leibler. We only considered sparse data about the activities of users and utilized both trajectories and geotagged photos. The experiments show that sparse location history is useful, however, that not all measures are suitable for that purpose. L_2 and L_∞ give significantly poorer results than the other methods. Results with crisp histograms are better than with fuzzy histograms. The proposed method is useful, not only for location-aware recommendation systems, but also for other applications that want to measure the similarity of visit histories.

Further studies would be required to research the effects of normalization, with possible log-scaling of frequencies, cosine distance and fuzzy modeling on the performance of the methods.

Bibliography

[1] Adomavicius, G., Tuzhilin, A., "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions", *IEEE Transactions on Knowledge and Data Egineering*, Vol. 17 Issue 6, 734 – 749, 2005.

[2] Alahakone, A. U., Ragavan, V., "Geospatial Information System for Tracking and Navigation of Mobile Objects", *Int. Conf. on Advanced Intelligent Mechatronics*, pp. 875 – 880, Singapore, July 2009.

[3] Albanese, M., Chianes, A., d'Acierno, A., Moscato, V., Picariello, A., "A Multimedia Recommender Integrating Object Features and User Behavior", *Multimedia Tools and Applications*, Vol. 50, Isssue 3, 563 – 585, December 2010.

[4] Almer, A., Stelzl, H., "Multimedia Visualization of Geoinformation for Tourism Regions Based on Remote Sensing Data", *Symposium on Geospatial Theory, Processing and Applications*, Ottawa, Canada, 2002.

[5] Ananthanarayanan, G., Haridasan, M., Mohomed, I, Terry, D., Chandramohan, A. T., "StarTrack: a Framework for Enabling Track-Based Applications", *Int. Conf. on Mobile Systems, Applications and Services*, pp. 207 – 220, Kraków, Poland, June 2009.

[6] Bacon, K., Dewan, P., "Towards Automatic Recommendation of Friend Lists", *Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing*, pp. 1 – 5, Crystal City, USA, November 2009.

[7] Bao, J., Zheng, Y., Mokbel, M. F., "Location-based and Preference-aware Recommendation Using Sparse Geo-social

Networking Data", *Int. Conf. on Advances in Geographic Information Systems*, pp. 199 – 208, Redondo Beach, USA, 2012.

[8] Barbeau, S., Labrador, M. A., Perez, A., Winters, P., Georggi, N., Aguilar, D., Perez, R., "Dynamic Management of Real-Time Location Data on GPS-Enabled Mobile Phones", *Int. Conf. on Mobile Ubiquitous Computing, Systems, Services and Technologies*, pp. 343 – 348, Valencia, Spain, October 2008.

[9] Barrat, A., Weigt, M., "On the Properties of Small-world Network Models", *European Physical Journal*, Vol. 13, pp. 547 – 560, 2000.

[10] Bellotti, V., Bogole, B., Chi, E. H., Ducheneaut, N., Fang, J., Isaacs, E., King, T., Newman, M. W., Partrighe, K., Price, B., Rasmussen, P., Roberts, M., Schiano, D. J., Walendowski, A., "Acitivity-based Serendipitous Recommendation with the Magitti Mobile Leisure Guide", *ACM Conference on Human Factors in Computing Systems*, pp. 1157 – 1166, Florence, Italy, April 2008.

[11] Bezdek, J. C., Ehrlich, R., Full, W., "FCM: The Fuzzy *c*-means Clustering Algorithm", *Computers & Geosciences*, Vol. 10, Issue 2-3, pp. 191 – 203, 1984.

[12] Biagioni, J., Krumm, J., "Days of Our Lives: Assessing Day Similarity from Location Traces, User Modeling, Adaptation, and Personalization", *Lecture Notes in Computer Science*, Vol. 7899, pp. 89 – 101, 2013.

[13] Birukov, A., Blanzieri, E., Giorgini, P., "Implicit: An agent-based Recommendation System for Web Search", *Int. Conf. on Autonomous Agents and Multi-Agent Systems*, pp. 618 – 624, Utrecht, The Netherlands, July 2005.

[14] Bohte, W., Maat, K. "Deriving and Validating Trip Purposes and Travel Modes for Multi-days GPS-based Travel Surveys: A Large-scale Application in the Netherlands", *Transport Research*, Part C 17, pp. 285-297, 2009.

[15] Bolici, F., "No Hotel in D.C.", *Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing*, pp. 1 – 6, Crystal City, November 2009.

[16] Burke, R., "Hybrid Recommender Systems: Survey and Experiments", *User Modelling and User-Adapted Interaction*, pp. 331 – 370, Vol. 12, Issue 4, November 2002.

[17] Cha, S.-H., "Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions", *Mathematical Models and Methods in Applied Sciences*, Vol. 4, Issue 1, pp. 300 – 307, 2007.

[18] Cha, S.-H., "Taxonomy of Nominal Type Histogram Distance Measures", *American Conf. on Applied Mathematics*, pp. 325 – 330, Cambridge, USA, March 2008.

[19] Cha, S.-H., Srihari, S. N., "On Measuring the Distance between Histograms", *Pattern Recognition*, Vol. 35, Issue 6, pp. 1355 – 1370, 2002.

[20] Chen, M., Xu, M. and Fränti, P., "A Fast O(N) Multiresolution Polygonal Approximation Algorithm for GPS Trajectory Simplification", *IEEE Transactions on Image Processing*, Vol. 21, Issue 5, pp. 2770 – 2785, 2012.

[21] Chen, M., Xu, M. and Fränti, P., "Compression of GPS Trajectories Using Optimized Approximation", *Int. Conf. on Pattern Recognition*, pp. 2180 – 3183, Tsukuba, Japan, November 2012.

[22] Chen, X., Pang, J., Xue, R., "Constructing and Comparing User Mobility Profiles for Location-based Services", *ACM Symposium on Applied Computing*, pp. 261 – 266, 2013.

[23] Chen, Y., Jiang, K., Zheng, Y., Li, Ch., Yu, N., "Trajectory Simplification Method for Location-based Social Networking Services", *Int. Workshop on Location Based Social Network*, Seattle, USA, November 2009.

[24] Chow, Ch.-Y. and Mokbel, M. F., "Trajectory Privacy in Location-based Services and Data Publication", *ACM SIGKDD Explorations Newsletter*, Vol. 13, Issue 1, pp. 19 – 29, June 2011.

[25] Clements, M., Serdyukov, P., de Vries, A. P., Reinders, M. J. T., "Personalised Travel Recommendation Based on Location Co -occurrence", *The Computing Research Repository*, 2011.

[26] de Pessemier, T., Minnaert, J., Vanhecke, K., Dooms, S., Martens, L., 2013. "Social Recommendations for Events", *ACM Conf. on Recommender Systems*, Hong Kong, China, October 2013.

[27] Ducheneaut, N., Partridge, K., Huang, Q., "Collaborative Filtering Is Not Enough? Experiments with a Mixed-Model Recommender for Leisure Activities", *Int. Conf. on User Modeling, Adaptation and Personalization*, pp. 295 – 306, Trento, Italy, June 2009.

[28] Fober, T., Hullermeier, E., "Similarity Measures for Protein Structures Based on Fuzzy Histogram Comparison", *IEEE Int. Conf. on Fuzzy Systems*, pp. 1 – 7, Barcelona, July 2010.

[29] Follin, J. M., Bouju, A., Bertrand, F., Boursier, P., "Management of Multi-Resolution Data in a Mobile Spatial Information Visualization System", *Int. Conf. on Web Information Systems Engineering*, pp. 92 – 99, Rome, Italy, December 2003.

[30] Fränti, P., Chen, J., Tabarcea, A., "Four Aspects of Relevance in Sharing Location-based Media: Content, Time, Location and Network", *Int. Conf. on Web Information Systems and Technologies*, pp. 413 – 417, Noordwijkerhout, The Netherlands, May 2011.

[31] Fränti, P., Tabarcea, A., Kuittinen, J., Hautamäki, V., "Location-based Search Engine for Multimedia Phones", *IEEE Int. Conf. on Multimedia and Expo*, pp. 558 – 563, Singapore, July 2010.

[32] Gali, N., Tabarcea, A. and Fränti, P., "Extracting Representative Image From Web Page", *Int. Conf. on Web Information Systems and Technologies*, Lisbon, Portugal, May 2015.

[33] Ge, Y., Xiong, H., Tuzhilin, A., Xiao, K., Gruteser, M., Pazzani, M. J., "An Energy-Efficient Mobile Recommender System", *ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 899 – 908, Washington, USA, July 2010.

[34] Gonzalez, P. A., Weinstein, J. S., Barbeau, S. J., Labrador, M. A., Winters, P. L., Georggi, N. L., Perez, R., "Automating Mode Detection for Travel Behaviour Analysis by Using Global Postioning Systems-enabled Mobile Phones and Neural Networks", *Intelligent Transport Systems*, Vol. 4, Issue 1, pp. 37 – 49, 2010.

[35] Göksedef, M., Sule G.-Ö., "Combination of Web Page Recommender Systems", *Expert Systems with Applications*, Vol. 37, Issue 4, pp. 2911 – 2922, April 2010.

[36] Gratz, P., Botev, J., "Collaborative filtering via epidemic aggregation in distributed virtual environments", *Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing*, pp. 1 – 9, Crystal City, USA, November 2009.

[37] Guc, B., May, M., Saygin, Y., Körner, C., "Semantic Annotation of GPS Trajectories", *Int. Conf. on Geographic Science*, Girona, Spain, May 2008.

[38] Guy, I, Jacovi, M., Perer, A., Ronen, I., Uziel, E., "Same Places, Same Things, Same People? Mining User Similarity on Social Media", *ACM Conference on Computer Supported Cooperative Work*, pp. 41 – 50, Savannah, USA, 2010.

[39] Haridasan, M., Mohomed, I., Terry, D., Chandramohan, A. T., Li, Z., "StarTrack Next Generation: A Scalable Infrastructure for Track-Based Applications", *OSDI*, pp. 409 – 422, Vancouver, Canada, October 2010.

[40] Horozov, T., Narasimhan, N., Vasudevan, V., "Using Location for Personalized POI Recommendation in Mobile Environments", *Int. Symposium on Applications and the Internet*, pp. 124 – 129, Phoenix, USA, January 2006.

[41] Ito, M., Nakazawa, J., Tokuda, H., "mPATH: An Interactive Visualization Framework for Behavior History", *Int. Conf. on Advanced Information Networking and Applications*, Taiwan, pp. 247 – 252, March 2005.

[42] Jakobs, K., Pils, C., Wallbaum, M., "Using the Internet in Transport Logistics - The Example of a Track & Trace System", *Int. Conf. on Networking*, pp. 194 – 203, Colman, France, July 2001.

[43] Kang, E.-Y., Kim, H., Cho, J., "Personalization Method for Tourist Point of Interest (POI) Recommendation", *Int. Conf. on Knowledge-Based Intelligent Information and Engineering Systems*, pp. 392 – 400, Bournemouth, UK, October 2006.

[44] Kim, Y., Cho, S.-B., "A Recommendation Agent for Mobile Phone Users Using Bayesian Behavior Prediction", *Int. Conf. on Mobile Ubiquitous Computing*, pp. 283 – 288, Sliema, Malta, October 2009.

[45] Kima, W., Jeong, O.-R.,Lee, S.-W., "On Social Web Sites", *Information Systems*, Vol. 35, Issue 2, pp. 215 – 236, April 2010.

[46] Lee, B.-H., Kim, H.-N., Jung, J.-G., Jo, G., "Location-based Service with Context Data for a Restaurant Recommendation", *Int. Conf. on Database and Expert Systems Applications*, pp. 430 – 438, Kraków, Poland, September 2006.

[47] Lee, W.-Ch., Krumm, J., "Computing with Spatial Trajectories", chapter 1, *Springer*, 2011.

[48] Lehtimäki, T., Partala, T., Luimula, M., Verronen, P., "LocaweRoute: An Advanced Route History Visualization for Mobile Devices", *Working Conf. on Advanced Visual Interfaces*, pp. 392 – 395, Napoli, Italy, May 2008.

[49] Levandoski, J. J., Sarwat, M., Eldawy, A., Mokbel, M.-F., "LARS: a Location-aware Recommender System", *Int. Conf. on Data Engineering*, pp. 450 – 461, Washington, USA, April 2012.

[50] Lew, M. S., Sebe, N., Djeraba, Ch., Jain, R., "Content-Based Multimedia Information Retrieval: State of the Art and Challenges", *ACM Transactions on Multimedia Computing, Communications and Applications*, Vol. 2, Issue 1, February 2006.

[51] Li, L.-H., Lee, F.-M. and Chen, Y.-C., "A Multi-stage Collaborative Filtering Approach for Mobile Recommendation", *Int. Conf. on Ubiquitous Information Management and Communication*, pp. 88 – 97,Suwon, South Korea, January 2009.

[52] Li, X., Guo, L., Zhao, Y., "Tag-based social interest discovery", *Int. Conf. on World Wide Web*, pp. 675 – 684, Beijing, China, 2008.

[53] Li, Q., Zheng, Y., Xie, X., Chen, Y., Liu, W., Ma, W.-Y., "Mining User Similarity Based on Location History", *ACM Int. Conf. on Advances in Geographic Information Systems*, Irvine, USA, November 2008.

[54] Liu, H., Schneider, M., "Similarity measurement of moving object trajectories", *Int. Workshop on GeoStreaming*, pp. 19 – 22, 2012.

[55] Liu, Y., Wilde, E., "Personalized Location-Based Services". *iConference*, pp. 496 – 502, Seattle, USA, 2011.

[56] Mariescu-Istodor, R., "Detecting User Actions in Mopsi", *Master Thesis, University of Eastern Finland*, Joensuu, 2013.

[57] Mariescu-Istodor, R., Tabarcea, A., Saeidi, R. and Fränti, P., "Low Complexity Spatial Similarity Measure of GPS Trajectories", *Int. Conf. on Web Information Systems and Technologies*, pp. 62 – 69, Barcelona, Spain, April 2014.

[58] Martín, S., Cristóbal, E. S., Gil, R., Díaz, G., Oliva, N., Castro, M., Peire, J., "Finding the Way: Services for a Multi-View and Multi-Platform Geographic Information System", Int. Conf. on

Web Information Systems and Technologies, pp. 267 – 270, Funchal, Portugal, May 2008.

[59] McCullough, A., James, P., Barr, S., "A Service Oriented Geoprocessing System for Real-Time Road Traffic Monitoring", *Transactions in GIS*, Vol. 15, Issue 5, pp. 651 – 665, 2011.

[60] Morris, S., Morris, A., Barnard, K., "Digital Trail Libraries", *ACM/IEEE Joint Conf. on Digital Libraries*, pp. 63-71, Tucson, USA, June 2004.

[61] Oliveira, M., Troped, P. J., Wolf, J., Matthews, C. E., Cromley, E. K., Melly, S. J., "Mode and Activity Identification Using GPS and Accelerometer Data", *Annual Meeting of the Transportation Research Board*, 2006.

[62] Park, M.-H., Hong J.-H., Cho, S.-B., "Location-based Recommender System Using Bayesian User's Preference Model in Mobile Devices", *Int. Conf. on Ubiquitous Intelligence and Computing*, pp. 1130 – 1139, Hong Kong, China, July 2007.

[63] Pu, P., Chen, L., "A User-Centric Evaluation Framework of Recommender Systems", *ACM Conference on Recommender Systems*, pp. 157 – 164, Barcelona, Spain, September 2010.

[64] Reddy, S., Mun, M., Burke, J., Estrin, D., Hansen, M., Srivastava, M., "Using Mobile Phones to Determine Transportation Modes", *ACM Transactions on Sensor Networks*, Vol. 6, Issue 2, pp. 1-27, February 2010.

[65] M. Rezaei and P. Fränti, Clustering Geo-Referenced Data on Maps, manuscript 2015 (submitted).

[66] Ricci, F., "Mobile Recommender Systems", *Information Technology & Tourism*, Vol. 12, Issue 3, pp. 205 – 231, 2011.

[67] Ross, C., Orr, E.S., Sisic, M., Arseneault, J.M., Simmering, M.G., Orr, R.R., "Personality and Motivations Associated with Facebook Use", *Computers in Human Behavior*, Vol. 25, Issue 2, pp. 578 – 586, March 2009.

[68] Rubner, Y., Tomasi, C., Guibas, L. J., "A Metric for Distributions with Applications to Image Databases", *IEEE Int. Conf. on Computer Vision*, pp. 59 – 66, Bombay, India, January 1998.

[69] Savage, N. S., Baranski, M., Chavez, N. E., Höllerel, T., "I'm Feeling LoCo: A Location Based Context Aware Recommendation System", *Advances in Location-Based Services*, pp. 37 – 54, 2012.

[70] Schilke S. W., Bleimann, U., Furnell, S. M., Phippen, A. D., "Multi-dimensional-personalisation for Location and Interest-based Recommendation", *Internet Research*, Vol. 14, pp. 379 – 385, December 2004.

[71] Schiller, J., Voisard, A., "Location Based Services", chapter 1, *Morgan Kaufmann Publishers Inc.*, San Francisco, USA, 2004.

[72] Simon, J. R., Gonzalez, D. R., Grande, C. F., Gomez, C. E., de la Llave, A. P., Lacalle, F. O., Permingeat, K. D. R., "NEMOS: Working towards the 'social' mobile phone", *Int. Conf. on Media / Expo*, pp. 1784 – 1788, New York City, July 2009.

[73] Strelkov, V. V., "A new similarity measure for histogram comparison and its application in time series analysis", *Pattern Recognition Letters*, Vol. 29, Issue 13, pp. 1768 – 1774, October 2008.

[74] Tabarcea, A., Wan, Z., Waga, K. and Fränti, P., "O-Mopsi: Mobile Orienteering Game Using Geotagged Photos", *Int. Conf. on Web Information Systems & Technologies*, pp. 300 – 303, Aachen, Germany, May 2013.

[75] Takabi, H., Joshi, J. B. D., Karimi, H. A., "A Collaborative k-anonymity Approach for Location Privacy in Location-based Services", *Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing*, pp. 1 – 9, Crystal City, Washington, November 2009.

[76] Takeuchi, Y., Sugimoto, M. "CityVoyager: An Outdoor Recommendation System Based on User Location History",

Int. Conf. on Ubiquitous Intelligence and Computing, pp. 625 – 636, China, September 2006.

[77] Troped, P. J., Oliveira, M. S., Matthews, C. E., Cromley, E. K., Melly, S. J., Craig, B. A., "Prediction of Activity Mode with Global Positioning System and Accelerometer Data", *Medicine and Science in Sports and Exercise*, Vol. 40, Issue 5, pp. 972 – 978, 2008.

[78] Tuan, C.-C., Hung, C.-F., and Kuei, T.-C., "Location Dependent Collaborative Filtering Recommendation System", *Int. Conf. on Future Network Technologies*, Qingdao, China, August 2011.

[79] Tung, H.-W., Soo, V.-W., "A Personalized Restaurant Recommender Agent Service for Mobile E-Service", *IEEE Int. Conf. on e-Technology, e-Commerce and e-Service*, pp. 259 – 262, Taiwan, March 2004.

[80] Waga, K., Tabarcea, A. and Fränti, P., "Context Aware Recommendation of Location-Based Data", *Int. Conf. on System Theory, Control and Computing*, pp. 1 – 6, Sinaia, Romania, October 2011.

[81] Waga, K., Tabarcea, A. and Fränti, P., "System for Real Time Storage, Retrieval and Visualization of GPS Tracks", *Int. Conf. on System Theory, Control and Computing*, pp. 1 – 5, Sinaia, Romania, October 2012.

[82] Wang, H., Calabrese, F., Di Lorenzo, G., Ratti, C., "Transportation Mode Inference from Anonymized and Aggregated Mobile Phone Call Detail Records", *Int. Conf. on Intelligent Transportation Systems*, pp. 318 – 323, Funchal, Portugal, September 2010.

[83] Wang, H., Liu, K., "User Oriented Trajectory Similarity Search", *ACM Int. Workshop on Urban Computing*, pp. 103 – 110, 2012.

[84] Watts, D., Strogatz, S., "Collective Dynamics of 'Small-world' Networks", *Nature*, Vol. 393, pp. 440 – 442, 1998.

[85] Wu, P., Ding, W., Mao, Z., Tretter, D., “Close & Closer: Discover Social Relationship from Photo Collections”, *Int. Conf. on Multimedia and Expo*, 1652 – 1655, New York City, July 2009.

[86] Xiao, X., Zheng, Y., Luo, Q., Xie, X., “Finding Similar Users Using Category-based Location History”, *Int. Conf. on Advances In Geographic Information Systems*, pp. 442 – 445, San Jose, USA, November 2010.

[87] Xu, Ch., Ji, M., Chen, W., Zhang, Z., “Identifying Travel Mode from GPS Trajectories through Fuzzy Pattern Recognition”, *Int. Conf. on Fuzzy Systems and Knowledge Discovery*, pp. 889 – 893, Yantai, China, August 2010.

[88] Yang, F., Wang, Z. M., "A Mobile Location-based Information Recommendation System Based on GPS and WEB 2.0 Services", *WSEAS Transactions on Computers*, Vol. 8, Issue 4, pp. 725 – 734, April 2009.

[89] Yang, X., Steck, H., Guo, Y., Liu, Y., “On Top-k Recommendation Using Social Networks”, *ACM Conf. on Recommender Systems*, pp. 67 – 74, Dublin, Ireland, September 2012.

[90] Ying, J. J.-C., Lee, W.-C., Tseng, V. S., “Mining Geographic-temporal-semantic Patterns in Trajectories for Location Prediction”, *ACM transactions on Intelligent System and Technology*, Vol. 5, Issue 1, December 2013.

[91] Yoon, H., Zheng, Y., Xie, X., Woo, W., "Social Itinerary Recommendation from User-generated Digital Trails", *Personal and Ubiquitous Computing*, Vol. 16, Issue 5, pp. 469 – 484, June 2012.

[92] Zhao, S., Grasmuck, S., Martin, J., “Identity Construction on Facebook: Digital Empowerment in Anchored Relationships”, *Computers in Human Behavior*, Vol. 24, Issue 5, pp. 1816 – 1836, September 2008.

[93] Zheng, V. W., Zheng, Y., Xie, X., Yang, Q., “Towards Mobile Intelligence: Learning from GPS History Data for Collaborative Recommendation”, *Artificial Intelligence Journal*, April 2012.

[94] Zheng, Y., Chen, Y., Li, Q., Xie, X., Ma, W.-Y., “Understanding Transportation Modes Based on GPS Data for Web Applications”, *ACM Transactions on the Web*, Vol. 4, Issue 1, January 2010.

[95] Zheng, Y., Wang, L., Zhang, R., Xie, X., Ma, W.-Y., “GeoLife: Managing and Understanding Your Past Life over Maps”, *Int. Conf. on Mobile Data Management*, pp. 211 – 212, Beijing, China, April 2008.

[96] Zheng, Y., Zhang, L., Ma, Z., Xie, X., and Ma, W.-Y., "Recommending friends and locations based on individual location history", *ACM Transactions on the Web*, Vol. 5, Issue 1, February 2011.

[97] Zheng, Y., Zhang, L., Xie, X., Ma, W.-Y., “Mining Interesting Locations and Travel Sequences from GPS Trajectories”, *Int. Conf. on World Wide Web*, pp. 791 – 800, Madrid, Spain, April 2009.

Paper I

K. Waga, A. Tabarcea, R. Mariescu-Istodor and P. Fränti,
"Real time access to multiple GPS tracks",
9th Int. Conf. on Web Information Systems & Technologies
(WEBIST'13),
293 – 299, Aachen, Germany, 2013.

Real Time Access to Multiple GPS Tracks

Karol Waga, Andrei Tabarcea, Radu Mariescu-Istodor and Pasi Fränti
Speech and Image Processing Unit, School of Computing, University of Eastern Finland, Joensuu, Finland
{kwaga, tabarcea, radum, franti}@cs.uef.fi

Keywords: Geographic information systems, GPS tracks, trajectory storage and visualization.

Abstract: Increasing availability of mobile devices with GPS receiver gives users the possibility to record and share a variety of location-based data, including GPS tracks. We describe a complete real-time system for acquisition, storage, querying, retrieval and visualization of GPS tracks. The main problems faced are how to store the data, how to access and how to visualize large amount of data. We propose to reduce the quantity of the data to be visualized, without affecting visualization quality. In order to achieve this, our system uses a fast polygonal approximation algorithm for different map scales along with a bounding box solution.

1. INTRODUCTION

Mobile devices with geo-positioning facilitate the acquisition of location-based data. This allows people to track their outdoor movements while performing physical exercises or when traveling. Companies can manage their geographical information in real-time (Martín et al., 2008) and track the movement of their own vehicles in order to solve problems such as fleet management (Jakobs et al., 2001) or traffic congestion (McCullough et al., 2011). The collected tracks are usually uploaded to an online system in order to be viewed, managed and analyzed. However, accessing and visualizing large amount of data is time consuming.

We present *MOPSI Routes*, a complete system for storage, retrieval and visualization of GPS tracks that overcomes the most common disadvantages of similar systems. For example, existing real-time web based systems, such as www.gmapgis.com and www.gpsvisualizer.com, do not have the possibility to plot large number of points and tracks on the map. In such cases, displaying becomes slow and visualizing overlapping tracks is difficult. Other solutions, such as TopoFusion (Morris et al., 2004), propose combining and intersecting GPS tracks in order to create trails and minimize the data needed to be displayed, although the goal, creating a GPS network of trails, is different. Our solution is to display all the recorded tracks in real time by reducing the number of points that are plotted. This is done by fast multi-resolution polygonal approximation algorithm described in (Chen et al., 2012), which achieves better approximation result than the existing competitive methods. Furthermore, we minimize the time needed for drawing by using a bounding box solution for plotting only the points that are visible to the user.

MOPSI Routes is available as a part of MOPSI services (cs.uef.fi/mopsi) and addresses the issues of storage, querying, retrieval and visualization of GPS tracks, first described in (Waga et al., 2012b). Users voluntarily upload their GPS tracks using our mobile application, which is available for most modern mobile operating systems (Android, Windows Phone, iOS and Symbian).

Similar research projects include GeoLife (Zheng et al., 2008), the system presented in (Alahakone et al., 2009) and StarTrack (Ananthanarayanan et al., 2009).

GeoLife (Zheng et al., 2008) is a project which focuses on visualization, organization, fast retrieval and understanding of GPS track logs. The main goal of the project is understanding people lives based on raw GPS data. The main contribution is visualizing GPS data over digital maps by indexing the GPS trajectories based on uploading behavior of users. Similarly to MOPSI Routes, tracks are searched using spatial range and time query.

The tool described in (Alahakone et al., 2009) is used for manipulating, integrating and displaying geographical referenced information. The main purposes for the tool are path planning and navigation of mobile objects. The tool can be used in several applications such as: tracking, fleet management, security management and industrial

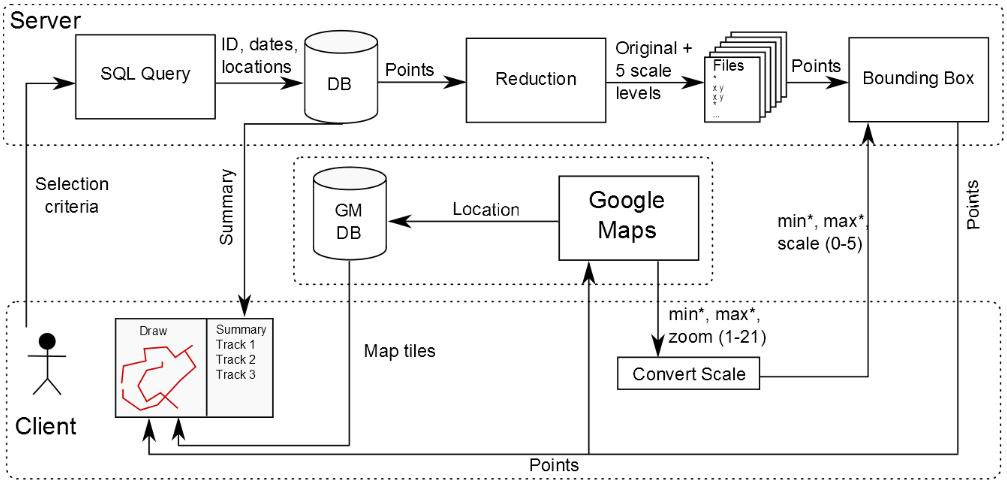


Figure 1. Typical workflow of MOPSI Routes.

robot navigation. Similarly to our system, a spatial database is used for storing tracks and points and Google Maps API to display the tracks. It presents a general approach in handling GPS data and it can be used in a variety of applications that use track recording, navigation and track planning. It requires that the user selects the points and defines the tracks, whereas our application automatically detects and segments the tracks.

StarTrack (Ananthanarayanan et al., 2009) and its improved version (Haridasan et al., 2010) describe tracks of coordinates as high-level abstraction for various types of location-based applications. The system supports recording, comparison, clustering and querying tracks. Experimental results show that the system is efficient and scalable up to 10.000 tracks. The improved version was extended to operate on collections of tracks, delay query executions and permit caching of query results. Other improvements are canonicalization based on road networks, and use of track trees for similarity.

2. SYSTEM DESCRIPTION

MOPSI Routes can be accessed at cs.uef.fi/mopsi/routes. The typical workflow of the system is presented in Fig. 1, whereas Fig. 2 shows example of tracks collection from one user.

In the first step, user selects the tracks to be displayed by several criteria such as time, location, duration and length. Tracks that match the criteria are retrieved from database and processed before displaying to the user. During the processing phase, the points belonging to the retrieved tracks undergo approximation process that reduces the number of points needed for the specific map scales. Points that are outside the visible area of the map are omitted by applying a bounding box. In the final step, the remaining points are shown on the map and the user can browse through them using map view (panning and zooming) or using list view to see additional information and statistics of each route.

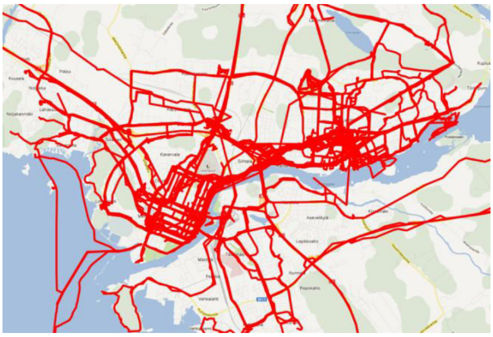


Figure 2. Example of user tracks collection.

2.1. Data Acquisition and Storage

MOPSI allows collecting tracking data using smartphones. The mobile application records the user's location and timestamp at a predefined interval (usually 1-4 seconds). The data is saved to database on server immediately if internet connection is available, or buffered on the device if internet connection is not available or the application is in offline mode.

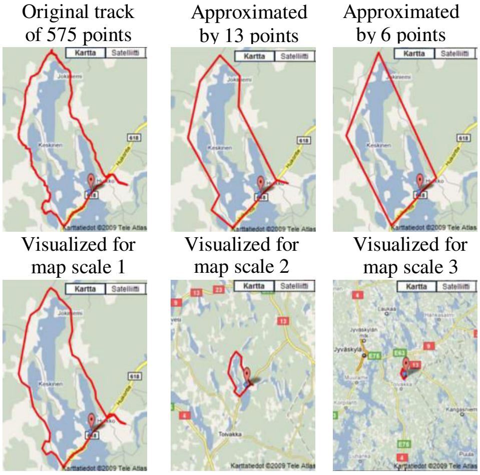
Tracks are first saved as individual points in the database, and track objects are created and updated real time when new points are received. Each track object contains not only the points but includes several basic statistics such as start and end time, bounding box and number of points. Segmentation and classification statistics are also stored. Analysis and classification of GPS tracks is described in details in (Waga et al., 2012a). Furthermore, each track is stored in its original and in a simplified form with reduced number of points. The approximated tracks are computed for 5 different zoom levels in order to speed up the visualization process. This limits the number of points drawn on the map without losing significant information about the shape of the GPS track. The analyzed and the approximated tracks are computed immediately when the points are uploaded.

The tracks are created and updated real time and tracking points are handled immediately after they have been uploaded. This process requires maintaining and updating track statistics and information constantly when user is recording a new track. To ensure this, there is a process running constantly on server that checks periodically (every 1 minute) if any track needs to be updated. When new tracking points are uploaded, they are either used for creating a new track object or merged with the existing points and inserted into list of the track's points in time order. The existing tracks are updated in the case that new tracking points belonging to an older track are received with significant delay caused, for example, by poor internet connection.

2.2. Different Map Scales

The tracks recorded in our system carry far more data than needed for visualization. Full data is needed for analysis, and therefore, complete GPS tracks must be stored. However, in the rendering process for a web browser, reduced number of points is sufficient to present the shape of track to user. We apply here a multi-resolution polygonal approximation algorithm described in (Chen et al., 2012). The algorithm is fast and achieves good quality approximation of the tracks. It is applied to every track and returns approximation of a track in 5 different map scales. The algorithm time complexity is O(N) (Chen et al., 2012) and the results are stored to avoid running algorithm repeatedly when the same track is displayed again.

Figure 3 shows an example of the original and approximated tracks. The original track contains 575 points and it is approximated in different map scales with 44, 13, and 6 points respectively. Suitable approximation error tolerance is selected for each map scale, and the visualization quality is not affected by the approximation, but rendering time is reduced significantly.


Original track of 575 points

Approximated by 13 points

Approximated by 6 points

Visualized for map scale 1

Visualized for map scale 2

Visualized for map scale 3

Figure 3. Visualization of a sample track.

2.3. Bounding Box

The purpose of the bounding box is to draw on the map only the points that are visible to user, see Fig. 4. Therefore, we select only points that user will see using the current map scale and location (*bounding box* of the map) at the moment of query. In addition, we draw also points that are outside the bounding box, but within immediate neighborhood (50% extension of screen size). In this way, we allow fast panning and zooming.

The bounding box is implemented as a function that gets coordinate of north, east, south and west of the map visible on the screen. Map scale is also passed, so that points from the correct approximation can be selected. The function is applied to every track and for every point it checks if the point lies inside the bounding box. Time complexity of the bounding box is linear and it is computed entirely on server.

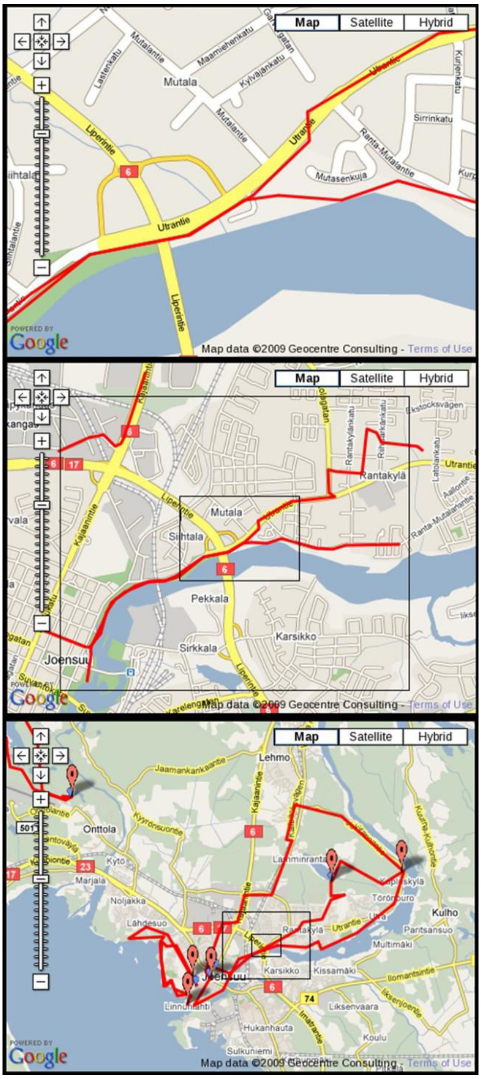
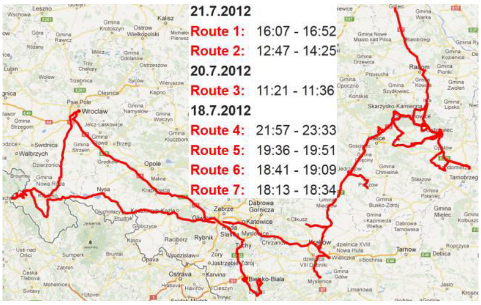


Figure 4. How the bounding box works (from top to bottom): what user sees on screen, what is drawn on map, all tracks selected.

2.4. Displaying Tracks on Map

In MOPSI, we use Google Maps to visualize the data (see Fig. 5). However, user can select different type of maps that are displayed as overlay over Google Maps. We support OpenStreetMaps and detailed orienteering maps in Joensuu area where most MOPSI users come from.

There are several search options available. The main search criterion is time, thus only tracks in the selected time period are shown. In addition, other criteria can be applied. For example, tracks can be filtered by minimum and maximum length and duration. Moreover, it is possible to search for tracks that start and end around a certain location.



21.7.2012
Route 1: 16:07 - 16:52
Route 2: 12:47 - 14:25
20.7.2012
Route 3: 11:21 - 11:36
18.7.2012
Route 4: 21:57 - 23:33
Route 5: 19:36 - 19:51
Route 6: 18:41 - 19:09
Route 7: 18:13 - 18:34

Figure 5. Displaying tracks on the map.

3. RESULTS

We measure the time spent between sending request to the system and presenting the result to the user. The time elapsed from user's query to the time of displaying the tracks on the screen using our system is compared with the same system that does not have reduction.

In all measurements, we ignore the time needed for data transfer. However, in weak internet access this might become bottleneck, and therefore, we design the system so that it minimizes data transfer. That allows using the system on computers with slower internet connection as well as on tablets that usually have limited bandwidth.

Table 1. Collections used for our experiments.

User	Tracks	Points	Length	Duration
			(km)	(h)
Pasi	784	1,216,039	8,535	669
Karol	650	1,015,939	9,655	442
Radu	429	613,684	4,604	188

We present measurements for 3 sample users from Table 1. The original tracks consist of large number of points. In MOPSI, there are users having over one million points, which shows the need for reducing the number of points being displayed as none of the browser could handle such large number of points (Chen et al., 2009). In Table 2, we show the number of points from the original tracks within the selected time period and the number of points from the approximated tracks. The zoom level of the map is selected in such manner that all the tracks are visible on the map.

Figure 6 presents the time needed to display tracks in a selected period for three test users. The process is divided into three phases: querying database, computing bounding box and drawing in browser. Results show that the time needed for showing all the tracks of the user with the biggest collection is about 2.5 seconds.

Table 2. Number of points in original (left) and in the approximated tracks (right) in the selected time period for user Pasi.

	Original	**Approximated**
all	1,216,039	9,064
year	424,709	3,088
month	46,669	331
week	11,204	903
recent	3,328	141

Figure 7 shows average time percentages spent in each of the three phases. Querying data takes most of the time. Calculating bounding box is a fast process that additionally speeds up drawing tracks on map, so that it takes only 14% of time.

The approximation algorithm is necessary to reduce the number of points displayed. Without it, it is not possible to display all tracks because the web browser would crash. The number of points browsers can handle depends on available resources. Displaying thousands of points significantly slows down web browsers. Nevertheless, even if browser can display all the points in tracks, the time needed for the process increases.

Table 3. Size of files (in bytes) with original and approximated tracks for user Karol.

	Original	Approximated
week	14.000	148
month	346.000	2280
year	4.056.000	69.000
all	11.595.000	129.000

Bigger number of points slows down the bounding box algorithm and often leads to memory issues. Moreover, approximation algorithm reduces files sizes as shown in Table 3 and preserves bandwidth used to retrieve data from server.

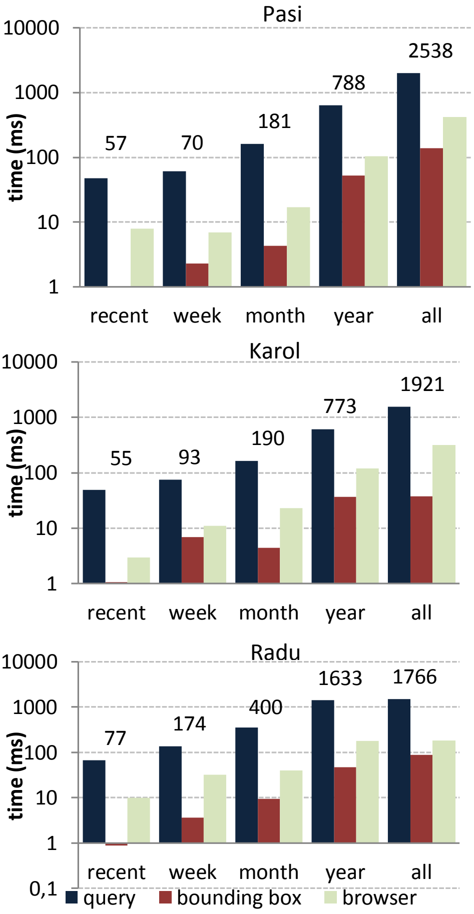


Figure 6. Display times of track collection for users Pasi, Karol and Radu.

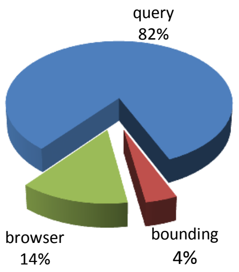


Figure 7. Average time percentage used for performing each operation of the system.

Experiments show that applying bounding box decreases time needed to draw tracks on map. Fig. 8 shows a sample case from the experiments. In this case the same set of tracks was requested at the same zoom level, but the map was focused in two different places, Finland and Poland. In Finland the collection of tracks is big, whereas in Poland there are only several tracks available. Because of applying the bounding box solution, not all the tracks have to be displayed and the time to show the tracks when map shows fewer tracks (Poland area) is significantly shorter. Figure 8 also shows how reducing number of points affects the display time.

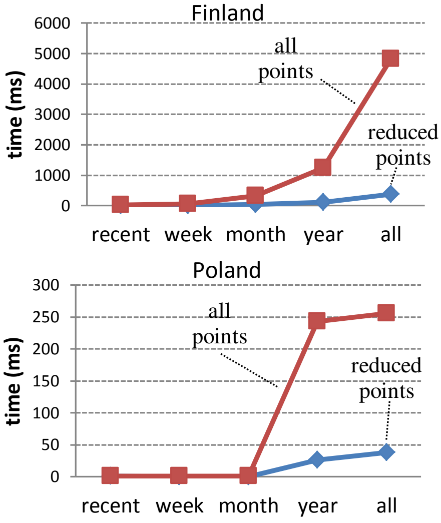


Figure 8. Example of querying the same track collection the same zoom level and focused in Finland (large collection, top) and Poland (small collection, bottom).

In comparison with the existing web based systems for visualizing GPS tracks, our system can display data consisting of significantly more points. For example, a track with about 10.000 points is displayed by our system in 1 second whereas GPS visualizer (www.gpsvisualizer.com) and GMapGis (www.gmapgis.com) need approximately 5 seconds. Moreover, user interaction is not slowed down in our system, when large number of points being is displayed.

4. SUMMARY

We presented a complete real time system to collect and visualize GPS tracks. Our motivation is to offer a system that is capable of handling large amount of GPS data so that user can access them in real time. The results show that our system is efficient even with large point collection. The most important part is the algorithm reducing the number of points to be displayed. Combined with a bounding box solution, the requested tracks can be accessed within about 2.5 seconds and the collection can be panned and zoomed with insignificant delay. The developed system can be used as a basis for more advanced analysis of GPS tracks, such as similarity and movement type detection.

Although, the system is efficient, there are still ways to improve it. For instance, now we reduce the number of points of one track only, but not when multiple tracks are overlapped. Further improvement could be achieved by clustering partial track segments. Moreover, the query phase should be optimized to minimize time needed to retrieve data.

REFERENCES

Alahakone, A. U., Ragavan, V. Geospatial Information System for Tracking and Navigation of Mobile Objects. ICAIM 09. Singapore, July, 2009.

Ananthanarayanan, G., Haridasan, M., Mohomed, I, Terry, D., Chandramohan, A. T. StarTrack: a Framework for Enabling Track-Based Application. ICMAS 09. Kraków, Poland, June 2009.

Chen, M., Xu, M., Fränti, P. A Fast O(N) Multi-resolution Polygonal Approximation Algorithm for GPS Trajectory Simplification. IEEE Trans. on Image Proc. 21(5). 2012.

Chen, Y., Jiang, K., Zheng, Y., Li, Ch., Yu, N. Trajectory Simplification Method for Location-based Social Networking Services. Int. Workshop on Location Based Social Network. Seattle, USA, November 2009.

Haridasan, M., Mohomed, I., Terry, D., Chandramohan, A. T., Li, Z. StarTrack Next Generation: A Scalable Infrastructure for Track-Based Applications, 2010.

Jakobs, K., Pils, C., Wallbaum, M. Using the Internet in Transport Logistics - The Example of a Track & Trace System. Networking ICN, 194-203, 2001.

Martín S., Cristóbal E.S., Gil R., Díaz G., Oliva N., Castro M., Peire J. Finding the Way: Services for a Multi-View and Multi-Platform Geographic Information System. WEBIST (2), pp.267-270, 2008.

McCullough, A., James, P., & Barr, S. (2011). A Service Oriented Geoprocessing System for Real-Time Road Traffic Monitoring. Transactions in GIS, 15(5), 651-665, 2011.

Morris, S., Morris, A., Barnard, K. Digital Trail Libraries. ACM/IEEE-CS Joint Conf. on Digital Libraries, pp. 63-71, June, 2004.

Waga, K., Tabarcea, A., Chen, M., Fränti, P. Detecting Movement Type by Route Segmentation and Classification. CollaborateCom, Pittsburgh, USA, October 2012.

Waga, K., Tabarcea, A., Mariescu-Istodor R., Fränti, P. System for Real Time Storage, Retrieval and Visualization of GPS Tracks. ICSTCC, Sinaia, Romania, October 2012.

Zheng, Y., Wang, L., Zhang, R., Xie, X., Ma, W.-Y. GeoLife: Managing and Understanding Your Past Life over Maps. Int. Conf. on Mobile Data Mgmt., Beijing, China, April 2008.

Paper II

K. Waga, A. Tabarcea, M. Chen and P. Fränti,
"Detecting movement type by route segmentation and classification",
8th IEEE Int. Conf. on Collaborative Computing: Networking,
Applications and Worksharing (CollaborateCom'12),
508 – 513, Pittsburgh, USA, 2012.

Detecting Movement Type by Route Segmentation and Classification

Karol Waga, Andrei Tabarcea, Minjie Chen, Pasi Fränti
Speech & Image Processing Unit, School of Computing
University of Eastern Finland, Joensuu
{karol.waga, andrei.tabarcea, minjie.chen, pasi.franti} @ uef.fi

Abstract—**Data about people movement is nowadays easy to collect by GPS technology embedded in smartphones. GPS routes provide information about position, time and speed, but further conclusion requires either prior information or data analysis. We propose a method to detect the movement type by segmentation of the GPS route using speed, direction and their derivatives, and by applying an inference algorithm via a second order Markov model. The method is able to classify most typical moving types such as motor vehicle, bicycle, run, walk and stop.**

Keywords: route analysis, segmentation, classification, GPS trajectory routes, tracks, mobile applications, second order Markov model.

1. Introduction and Motivation

Mobile phones have developed rapidly and most people are using one nowadays. Many of the phones, especially smartphones, are equipped with multiple sensors including global positioning system (GPS). This makes it possible to implement various location-aware services varying from recommendation of point-of-interests [9] to sport activity tracking. GPS data, however, can capture only features such as speed, distance, location and time which cannot be used to conclude semantic meaning of the user activity [5].

In this work, we aim at providing higher-level information of the user activity by detecting five most typical movement types: *stop*, *walk*, *run*, *bicycle*, and *motor vehicle*. The proposed method is based on route segmentation and a simple rule-based classifier as follows.

First, segmentation is performed in order to divide the analyzed route into a number of segments, where the sum of the inner speed variance is minimized. A set of basic features, such as speed, acceleration, time, direction and distance are then extracted for each segment. In the second stage, the extracted segments are classified into a predefined set of five classes using a second order Markov model.

The proposed method is implemented and tested using the data collected by users of MOPSI (in Finnish **MO**biilit **P**aikkatieto**S**ovellukset ja **I**nternet, Mobile Location-based Application and Internet) services (see Fig. 1). It implements location-aware services such as user tracking, route recording, photo collection, recommendations, bus schedules and sharing data in Facebook. All these are available through website and mobile platforms in Symbian, Android, iPhone and Windows Phone devices. The proposed movement type detection algorithm includes three steps that are preprocessing, segmentation and classification. They are implemented in C language and executed on server real-time when user requests route analysis.

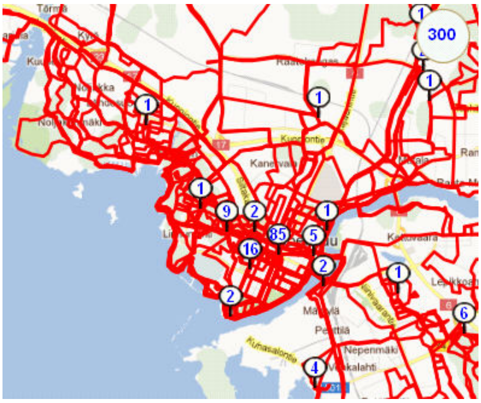


Figure 1. Sample data collected in MOPSI route system: http://cs.uef.fi/mopsi/

We demonstrate by qualitative tests that the proposed method can successfully detect most typical movement types based on raw GPS data only. Better accuracy would require the use of other sensors such as phone accelerometer, heart rate, or using external information such as road network, bus stops and bus schedules. Geographic information could be used for separating between land and water although not enough to separate boating from cross-country skiing on lake.

2. Related work

Most work on the automatic determination of travel mode uses GPS data, but few attempts exist to use other types of data including call detail records [10] and cellular network positioning data. A study on different combinations of GPS, GSM, WiFi and accelerometer was constructed in [8]. It concludes that the combination of GPS and accelerometer is the best data source for the travel model detection.

A simple approach is to measure the speed of the GPS device, which is then compared with empirical thresholds [1, 7]. However, some transport modes, such as cycling and running, are hard to differentiate using only the speed thresholds. More complex solutions have therefore been considered, based on methods such as *fuzzy logic* [11], *neural networks* [4] and *hidden Markov model* [8, 13].

Xu et al. [11] proposed travel mode detection using fuzzy logic with five movement types: *walk, bike, bus, rail* and *rest*. The routes are divided into stages using fuzzy pattern recognition and a min-max operation. The variables used for fuzzy calculations are median speed, average speed, standard deviation of speed and minimum acceleration. Membership functions are created for each fuzzy variable and the travel mode is detected with satisfying accuracy, except rail.

Using neural networks in travel mode classification was proposed in [4]. Adaptive sampling rate (1s rate outdoor and lower rate indoor) and on-device route reduction called *critical point algorithm* are used to conserve energy. Classification uses the following attributes: average speed, maximum speed, estimated horizontal accuracy uncertainty, percentage of GSM fixes compared to GPS, standard deviation of distances between stop locations, average dwell time (how much does a stop take). The neural network uses just the critical points to classify a route as car, walk or bus and the accuracy of this method is 80-90%.

A three-stage approach was proposed in [13] consisting of the following steps: segmentation by change point detection, feature selection followed by classification using an inference model and graph-based probabilistic post-processing. The advantages of the approach are that it can effectively segment routes with various transportation modes, it is dependent only on GPS data (no additional sensor data or map information) and the model learned from some users can be applied to infer GPS data from other users. However, in most of the cases, the change points are considered to be walk points. Therefore, only non-moving or walking parts of the trajectory are detected as segment boundaries. Additionally, we consider speed changes as segment boundaries, having different segmentation results, for instance when the user travels from highway to urban areas by car.

In [8], a decision tree followed by a first-order hidden Markov model is used. This method has 93.6% accuracy and it does not rely on GIS information or historical data as in [13]. The paper also compares different classifiers: decision trees, k-means clustering, naïve Bayes, nearest neighbor, support vector machines, continuous hidden Markov model, and a decision tree combined with a discrete hidden Markov model. The decision tree classifier provided the most accurate result of the simple classifiers, and its combination with discrete hidden Markov model was the most accurate overall.

These have two main weaknesses that we address. Firstly, training material is not always available, and may result in over-fitting, which loses generalization. Secondly, the correlation between neighboring segments is not fully exploited and depends on the segmentation accuracy. By using second-order Markov model we address this problem and we benefit from the correlation to the previous and also to the next segment.

3. Proposed Solution

The challenge of the movement type detection is demonstrated in Fig. 2 and Fig. 3.

The first example in Fig. 2 includes interval training with three faster segments separated by shorts stops, and two slower jogging periods in the beginning and in the end. The challenge here is to accurately detect the stop points. The labels in these pictures are ground truth provided by the person who performed the exercise.

The second example in Fig. 3 includes three fast downhill skiing segments, two slow uphill movements by elevator, and two waiting periods (lining up in queue waiting to access the skiing elevator). Although it is not possible to conclude all this activity merely from the GPS data, segmentation can help to obtain better classification.

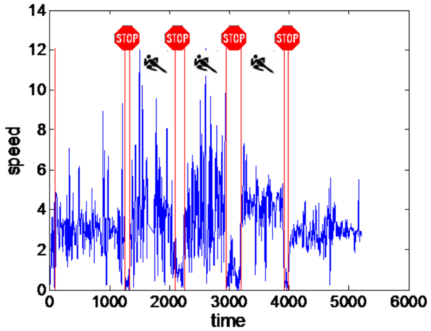


Figure 2. Non-trivial example of movement type analysis: *interval training.*

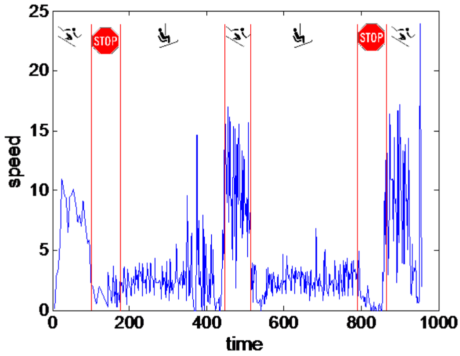


Figure 3. Non-trivial example of movement type analysis: *quality skiing time in elevator.*

A. Route Segmentation

We first divide a route into segments with similar speed. The number of segments is automatically determined but could also be predefined by user.

An input to the algorithm is a route $P = (p_1, p_2,...,p_n)$, where $p_i= (x_i, y_i, t_i)$, and the corresponding speeds are $(v_1, v_2,...,v_{n-1})$. For a given segment number m, we define a cost function that minimizes the sum of the inner speed variance in all the segments:

$$f = \sum_j \sigma_{i_j}^{i_{j+1}} (t_{i_{j+1}} - t_{i_j}) \qquad (1)$$

where i_j and i_{j+1} are the indexes of the start and end points of the segment j, and σ is the speed variance between the points j to j+1 in route p_i. Our experiments have shown that the proposed cost function is more efficient than mean square error, which has difficulties to detect walking segments with lower speed.

This minimization process is solved by a dynamic programming process in O(n^2m) time and O(nm) space, where the speed variance can be calculated in O(1) time by using the pre-calculated accumulated sums. Optimization is done in the $n\times m$ state space using dynamic programming as follows:

$$\begin{aligned} D(s,r) &= \min(D(c,r-1) + \sigma_c^s(t_s - t_c)), c \in (1...s-1) \\ A(s,r) &= \arg\min_c (D(c,r-1) + \sigma_c^s(t_s - t_c)) \end{aligned} \qquad (2)$$

where $s = 0\ldots n$, $r = 0\ldots m$, with an initial condition $D(0, 0) = 0$, and $A(s, r)$ is the index for backtracking. The number of segments m_0 is determined by

$$m_0 = \arg\min_i (D(n,i) + \lambda_1 i + \lambda_2 (t_n - t_1)), i = 1...m \qquad (3)$$

where λ_1, λ_2 are regularization parameters.

B. Moving Type Classification

Several features such as speed, acceleration, time, direction and distance can be calculated. However, training a classifier directly on these features is not accurate, since many features overlapped over different movement types [13]. Instead, we first perform soft classification of each segment as *stop*, *walk*, *run*, *bicycle* or *motor vehicle* using the a priori probabilities shown in Fig. 4.

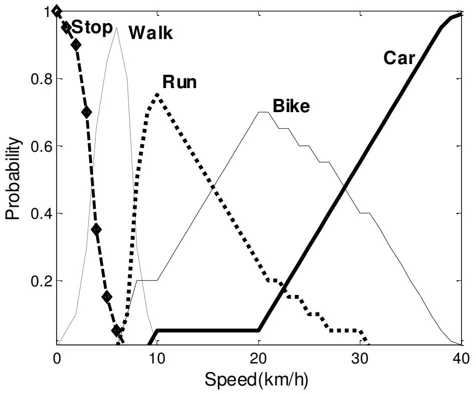


Figure 4. A priori probabilities for soft classification of the route segments.

On the other hand, the first order hidden Markov model (HMM) has been successfully used to exploit correlations between neighboring segments [8]. In this model, the hidden states represent the movement types and the observed data are the features for each segment. We extend this to a second-order HMM by exploiting correlations both to the previous and the next segment.

The state transition matrix is empirically initialized as shown in Table 1, but it could also be optimized via a training process in further stages of the application if training data exists.

Table 1: Transition probabilities used in the 2-HMM

Prev.	Probability:					Next
	🚗	🚲	🏃	🚶	🛑	
🚗	0.6	-	-	0.2	0.2	🚗
🚗	0.5	0.2	-	0.1	0.2	🚲
🚗	0.5	-	0.2	0.1	0.2	🏃
🚗	0.5	-	-	0.3	0.2	🚶
🚗	0.8	-	-	0.1	0.1	🛑
🚲	0.5	0.2	-	0.1	0.2	🚗
🚲	-	0.6	-	0.2	0.2	🚲
🚲	-	0.4	0.4	0.1	0.1	🏃
🚲	-	0.4	-	0.4	0.2	🚶
🚲	-	0.8	-	0.1	0.1	🛑
🏃	0.5	-	0.2	0.1	0.2	🚗
🏃	-	0.4	0.4	0.1	0.1	🚲
🏃	-	-	0.4	0.4	0.2	🏃
🏃	-	-	0.4	0.4	0.2	🚶
🏃	-	-	0.8	0.1	0.1	🛑
🚶	0.5	-	-	0.3	0.2	🚗
🚶	-	0.4	-	0.4	0.2	🚲
🚶	-	-	0.4	0.4	0.2	🏃
🚶	0.2	0.2	0.2	0.2	0.2	🚶
🚶	-	-	0.1	0.7	0.2	🛑
🛑	0.8	-	-	0.1	0.1	🚗
🛑	-	0.8	-	0.1	0.1	🚲
🛑	-	-	0.8	0.1	0.1	🏃
🛑	-	-	0.1	0.7	0.2	🚶
🛑	0.2	0.2	0.2	0.2	0.2	🛑

For the cost function of the 2nd order HMM we use:

$$f = \prod_{i=1}^{M} P(m_i \mid \mathrm{X}_i, m_{i-1}, m_{i+1}) \qquad (4)$$

where m_i = {*stop*, *walk*, *run*, *bicycle* or *motor vehicle*} is the state of segment *i*, X_i is its feature vector, m_{i-1}, m_{i+1} are the states of the previous and the next segment. The probability that a segment would have a hidden state m_i depends on the previous state, the next state and its feature vector. After Eq. (4) has been maximized, the most likely sequence of the hidden state m_0, $m_1 \ldots m_M$ is determined.

Assuming the feature vector X_i is uncorrelated with m_{i-1} and m_{i+1}, this cost function can be converted by applying Bayesian inference:

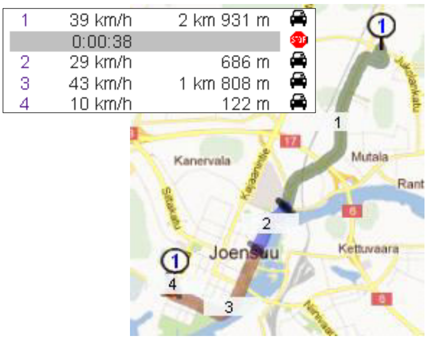
$$f = \prod_{i=1}^{M} \frac{P(m_{i+2} \mid m_i, m_{i+1}) P(m_{i+1} \mid X)}{P(m_{i+2})} \qquad (5)$$

where $P(m_{i+2}|m_i, m_{i+1})$, $P(m_i \mid \mathrm{X}_i)$ and $P(m_i)$ are all prior information. In the implementation of the algorithm, dynamic programming is employed for maximizing the cost function (5). The maximizing is done in a similar manner as Viterbi algorithm, which has been used for the first order HMM.

4. Experiments

The proposed system is implemented in Mopsi, which is publicly available (http://cs.uef.fi/mopsi). Segmentation and travel mode classification are triggered via *Analyze* function in the routes view. After analysis, the statistics of the segments (speed, distance and movement type) are displayed. In Mopsi, there are more than 7000 routes that in total have more than 4 million points, which are selected here as qualitative examples to demonstrate the capability and limitations of the algorithm.

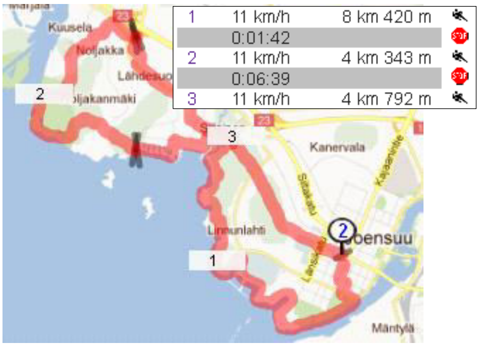
Fig. 5 shows a car tour with one stop at traffic lights. The speed segments demonstrate typical traffic flow in Joensuu. The last segment is classified also as motor vehicle based on the overall pattern despite its slower speed.



1	39 km/h	2 km 931 m	🚗
	0:00:38		🛑
2	29 km/h	686 m	🚗
3	43 km/h	1 km 808 m	🚗
4	10 km/h	122 m	🚗

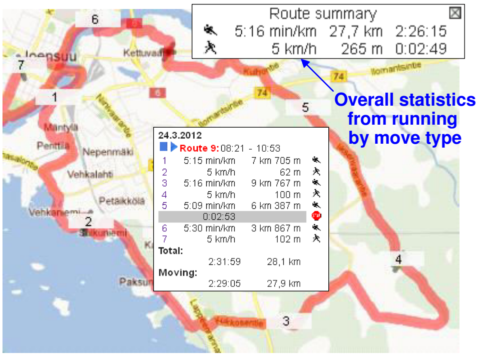
Figure 5. Segmentation of a car route with one stop.

Fig. 6 and 7 show sport exercises where the running speed can be easily concluded from the moving segments, even though one short stop (segment 2) is incorrectly classified as walking.



1	11 km/h	8 km 420 m	🏃
	0:01:42		STOP
2	11 km/h	4 km 343 m	🏃
	0:06:39		STOP
3	11 km/h	4 km 792 m	🏃

Figure 6. Separating stop segments from running.



Route summary

🏃	5:16 min/km	27,7 km	2:26:15
🚶	5 km/h	265 m	0:02:49

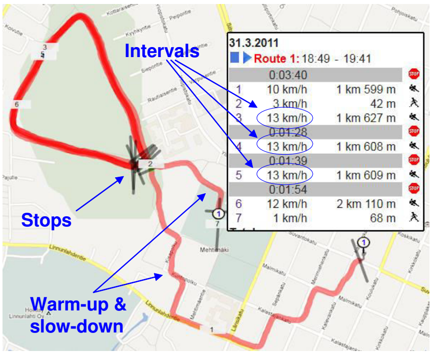
24.3.2012

Route 9: 08:21 - 10:53

1	5:15 min/km	7 km 705 m	🏃
2	5 km/h	62 m	🚶
3	5:16 min/km	9 km 767 m	🏃
4	5 km/h	100 m	🚶
5	5:09 min/km	6 km 387 m	🏃
	0:02:53		STOP
6	5:30 min/km	3 km 867 m	🏃
7	5 km/h	102 m	🚶
Total:			
	2:31:59	28,1 km	
Moving:			
	2:29:05	27,9 km	

Figure 7. Long-distance running.

In Fig. 8, three interval exercises are captured, along with warm-up and slow-down. The classifier recognizes the exercises correctly as running segments with similar average speed (13 km/h). The segmentation itself makes the analysis also easier for the user.



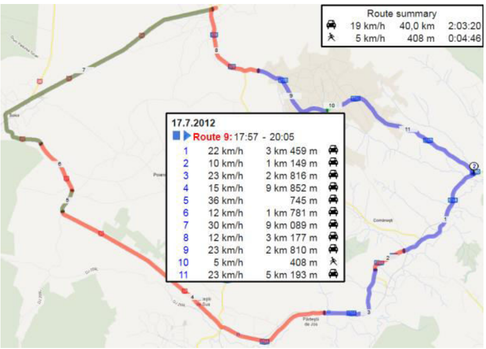
31.3.2011

Route 1: 18:49 - 19:41

	0:03:40		STOP
1	10 km/h	1 km 599 m	🏃
2	3 km/h	42 m	🚶
3	13 km/h	1 km 627 m	🏃
	0:01:28		STOP
4	13 km/h	1 km 608 m	🏃
	0:01:39		STOP
5	13 km/h	1 km 609 m	🏃
	0:01:54		STOP
6	12 km/h	2 km 110 m	🏃
7	1 km/h	68 m	🚶

Figure 8. Interval training exercise

The cycling segments in the route in Fig. 9 are incorrectly classified as motor vehicle. Despite the accurate segmentation and the correct detection of walking segment, the inaccuracies of GPS signal and high top speed cause incorrect classification. One segment classified as motor vehicle movement increases the probability of similar segments to be classified with the same transportation mode.



Route summary

🚗	19 km/h	40,0 km	2:03:20
🚶	5 km/h	408 m	0:04:46

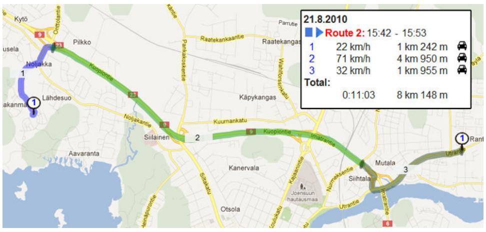
17.7.2012

Route 9: 17:57 - 20:05

1	22 km/h	3 km 459 m	🚗
2	10 km/h	1 km 149 m	🚗
3	23 km/h	2 km 816 m	🚗
4	15 km/h	9 km 852 m	🚗
5	36 km/h	745 m	🚗
6	12 km/h	1 km 781 m	🚗
7	30 km/h	9 km 089 m	🚗
8	12 km/h	3 km 177 m	🚗
9	23 km/h	2 km 810 m	🚗
10	5 km/h	408 m	🚶
11	23 km/h	5 km 193 m	🚗

Figure 9. Bicycle route classified as car.

Finally, Fig. 10 shows that our segmentation algorithm can detect speed changes as segment boundaries, even though the transportation mode is the same (car),



21.8.2010

Route 2: 15:42 - 15:53

1	22 km/h	1 km 242 m	🚗
2	71 km/h	4 km 950 m	🚗
3	32 km/h	1 km 955 m	🚗
Total:			
	0:11:03	8 km 148 m	

Figure 10. Travelling by car on highway between urban areas.

5. Discussion and Conclusions

The proposed approach was demonstrated to be useful for analyzing collected GPS trajectories and detecting typical movement types such as motor vehicle, bicycle, run, walk and stop. In some cases, segmentation missed short stops, and separation between bicycle and running would require user-specific information or data from accelerometer.

As a future work, we plan to implement annotation tool using the model in [5] to collect ground truth and perform numerical comparison with other methods.

References

[1] Bohte, W., Maat, K. Deriving and validating trip purposes and travel modes for multi-days GPS-based travel surveys: A large-scale application in the Netherlands. *Transport Research*, Part C 17, p. 285-297. 2009.

[2] Chen, M., Xu, M., Fränti, P. Compression of GPS Trajectories. *Data Compression Conference*. Snowbird, USA, 62-71, April 2012.

[3] Fränti, P., Tabarcea, A., Kuittinen, J., Hautamäki, V. Location-based search engine for multimedia phones. *IEEE Int. Conf. on Multimedia & Expo* (ICME'10), 558-563, Singapore, July 2010.

[4] Gonzalez, P.A., Weinstein, J.S., Barbeau, S.J., Labrador, M.A., Winters, P.L., Georggi, N.L., Perez, R. Automating Mode Detection for Travel Behaviour Analysis by Using Global Postioning Systems-enabled Mobile Phones and Neural Networks. *IET Intelligent Transportation Systems,* 4 (1), 37-49. 2010.

[5] Guc, B. Semantic Annotation of GPS Trajectories. *International Conference on Geographic Information Science*. Park City, Utah, USA, September 2008.

[6] Lee, W.-Ch., Krumm, J. Chapter 1: Trajectory Preprocessing, in *Computing with Spatial Trajectories*, Springer, 2011

[7] Oliveira, M., Troped, P. J., Wolf, J., Matthews, C. E., Cromley, E. K., Melly, S. J. Mode and Activity Identification Using GPS and Accelerometer Data. *Transportation Research Record*, 2006.

[8] Reddy, S., Mun, M., Burke, J., Estrin, D., Hansen, M., Srivastava, M. Using Mobile Phones to Determine Transportation Modes. *ACM Transactions on Sensor Networks*, 6 (2), 1-27, 2010.

[9] Waga, K., Tabarcea, A., Fränti, P. Context Aware Recommendation of Location-based Data. *Int. Conf. on System Theory, Control and Computing.* Sinaia, Romania, October 2011.

[10] Wang, H., Calabrese, F., Di Lorenzo, G., Ratti, C. Transportation Mode Inference from Anonymized and Aggregated Mobile Phone Call Detail Records. *Int. Conf. on Intelligent Transportation Systems,* Madeira, Portugal, September 2010.

[11] Xu, Ch., Ji, M., Chen, W., Zhang, Z. Identifying Travel Mode from GPS Trajectories through Fuzzy Pattern Recognition. *Int. Conf. on Fuzzy Systems and Knowledge Discovery,* Yantai, China, August 2011.

[13] Zheng, Y., Chen, Y., Li, Q., Xie, X., Ma, W.-Y. Understanding Transportation Modes Based on GPS Data for Web Applications, *ACM Trans. on Web*, 4 (1), 2010.

Paper III

K. Waga, A. Tabarcea and P. Fränti,
"Recommendation of points of interest from user generated data collection",
8th IEEE Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom'12),
550 – 555, Pittsburgh, USA, 2012.

Recommendation of Points of Interest from User Generated Data Collection

Karol Waga, Andrei Tabarcea, Pasi Fränti
Speech & Image Processing Unit, School of Computing, University of Eastern Finland, Joensuu
{kwaga, tabarcea, franti}@ cs.uef.fi

***Abstract*—Systems that aim to predict user preferences and give recommendations are now commonly used in many systems such as online shops, social websites, and tourist guides. In this paper, we present a context aware personalized recommendation system on web and mobile, which recommends relevant location-based data from user collection and consisting of GPS routes and photos. We recommend three types of items: services, photos and GPS routes that are points of interests in user's surrounding. We score all items from database based on four aspects of relevance: location, content, time and network. In order to personalize the results we built user profile based on user's activity in the system. We study performance of the system within MOPSI.**

Keywords—Recommendation, relevance, user collection, GPS trajectories, routes, context aware computing, location based systems

I. INTRODUCTION

A vast availability of location-acquisition devices and technologies (smartphones, GPS, GSM networks, mobile internet) allows people to record their activity by taking photos and tracking [16], especially if the service offers the possibility of sharing them with friends or with people who share the same interests. Gathering such data about activities of users allows discovering patterns of users' movements as well as information about points of interest in the area.

Recommendation systems produce personalized search results relying on a variety of contextual information. We have designed a recommendation system based on the four aspects of relevance: content, time, location and social network discussed in [3]. The system recommends items from a user-generated location-based dataset which consists of geotagged photos, trusted services and routes. Current version of the recommendation system is based on an earlier prototype as described in [12]. The goal of the recommendation is to suggest to user in certain location at given time where to go next, considering three types of items: services, photos and routes. Our solution is implemented within MOPSI system. The system includes various location-based services and applications such as search engines, data collection, user tracking and route recording. It has applications integrated both on web and in mobile phones. MOPSI contains services, photos and routes databases. Two latter ones are collected by service users utilizing a mobile application. The collections are shown in Fig. 1. Our user profile database used for giving personalized recommendation contains data about activities of MOPSI users within the service.

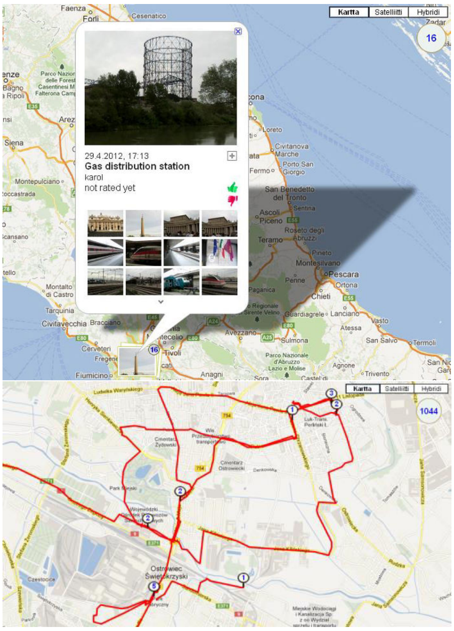

29.4.2012, 17:13
Gas distribution station
karol
not rated yet

Figure 1: Examples of photos and routes from MOPSI user collection

II. RELATED WORK

Using location based-based data and user's location as input to a context-aware recommendation algorithm has been studied in several research projects.

For example, the system described in [8] recommends online content and offline events based on current user location, adding location to multi-dimensional personalization. Events are also recommended in [6], which employs a multi-stage collaborative filtering process. Our system also recommends location-based data and considers user profiles and personalization, but the data is user-generated (photos and routes).

CityVoyager, a system described in [9], proposes recommendation of shops based on user location history. Visited locations (shops) are used as input to an item-based collaborative filtering algorithm. Similarly, we use location history as a relevance criterion.

Magitti, described in [1], predicts user activity from sensing context and from patterns of user behavior. The recommendation of content is generated automatically using a combination of various models, including collaborative filtering, distance, stated or learned preferences. We also recommend the content automatically, with minimum input from users.

The system in [16] recommends friends and places using individual location history. In the recommendation process, three factors are considered: user's particular location history, similarity between users in terms of location history, estimation of user's individual interests in an unknown region by comparing the location history and interest of other users. The collaborative filtering and recommendation algorithm is described in [15].

The service proposed in [13] uses web 2.0 technologies along with location-based services. A multi-dimensional collaborative filtering algorithm is designed in order to achieve dynamic personalized information which is delivered to mobile devices.

Location-Aware Recommender System (LARS) [5] uses location-aware ratings for recommendation. The ratings and items are considered to be spatial or non-spatial, also using the travel penalty approach which favors recommendation candidates close to user's location. The datasets used are social data from Foursquare and a part of MovieLens movie recommendation data. We also have a ranking-based system in which travel distance from user's position is an important factor.

"I'm feeling LoCo" [7] proposes a ubiquitous location-based recommendation algorithm that focuses on user experience by considering user preferences, time, location and similarity measures automatically, having Foursquare as a dataset. We also focus on user experience and aim that user input is minimal. The information from the user's social network, form of transportation and phone's sensors is inferred to provide recommendation of places from the dataset.

Recommending GPS trajectory is one of the main enhancements of the system described in [12]. GPS trajectory data mining is also the focus of [14], which extracts attributes from GPS trajectories in order to extract collective intelligence and recommend itineraries.

Another example is the system documented in [2], which recommends tourist locations based on user's visiting history in a geographically remote region. A set of geotags is used to compute location similarity and novel places are recommended to the user.

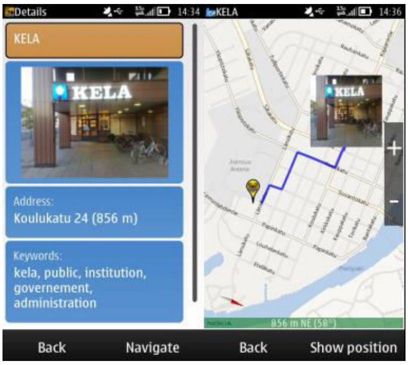
The agent system based described in [4] is based on prediction of the user's future behavior. The system understands the context from the GPS receiver and the prediction is performed by Dynamic Bayesian Networks. Finally, [10] proposes location-dependent collaborative filtering recommendation by using mobile user's location history and behavior prediction.

III. SYSTEM DESCRIPTION

In this section, we provide description of what are the typical use case scenarios, what our system actually recommends, and how it uses the four aspects of relevance identified in [3] as the recommendation context.

A. *Use case scenarios*

The work reported in this paper is part of MOPSI system. It implements various location-based services and applications such as search engines, data collection, user tracking and route recording. We have had so far 245 individuals using it. The recommendation system is available both on mobile device and on desktop computer. Mobile user interface is shown on Fig. 2 and web interface is shown on Fig. 3.


Details 14:34

KELA

Address:
Koulukatu 24 (856 m)

Keywords:
kela, public, institution, governement, administration

KELA 14:36

856 m NE (58°)

Figure 2: Recommendation results in mobile application. Left screenshot shows details of one of the recommended items. Right screenshot presents navigation screen to the selected item.

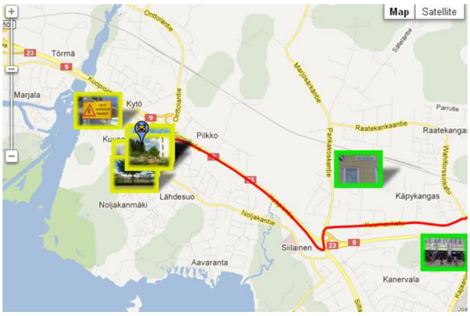


Figure 3: Recommendation system results on website.

The most common use case scenario of our system is user asking for recommendation using mobile device when the user is in a location he or she is not familiar with. Mobile access to the service is important since this is the most natural environment where the system can be most beneficial in real life. The key functionality of the system is that the user can ask for recommendation in any location so that he or she can easily visit suggested places immediately. The recommendation system suggests what the user can see and do in the surroundings. The results are displayed as list with the most recommended item on top. Depending on the type of recommended item, there are various additional statistics. The location of all items is also shown on map.

We provide recommendations also on website. In this case, usage scenario is that users search for what is interesting in the area before visiting it.

Access to the recommendation results requires connection to server, where the computations are done. In order to provide recommendation results immediately we calculate them in advance. Results are recalculated every time location of the user changes. The decision whether to recalculate or not a new set of recommended items is made on server-side by the recommendation synchronizer (see Fig. 4). The decision is based on the difference between user's current location and location in which recommendation was given previously to the user. Moreover, if user is in the same place, the time that passed since last recommendation is considered to assure that results are up to date.

B. *Recommended items*

MOPSI contains three databases that are used as a source for recommended items.

The first database contains trusted services verified by administrators illustrated by the green markers on the list and map in Fig. 3. These services represent variety of categories from restaurants, bars, and cafeterias, through grocery stores, pharmacies, and ATM machines, to car repairs, and museums. Service data include location, contact information, and relevant keywords as well as rating given by users.

The second database (user photo collection) contains photos users have taken using mobile phones and uploaded with several related information, such as location, time, and description as well as rating given by other users. Example of such collection is shown on top of Fig.1. In recommendation results these items are shown as yellow markers on the list and map in Fig. 3. MOPSI users collected 12.095 photos (6.8.2012).

The third database contains routes that users have recorded by mobile application. A route collection sample is shown at the bottom of Fig. 1. In recommendation results the routes are presented as red markers on the list and red lines on map shown in Fig. 3. Therefore it contains information about users' movements and places they have visited.

A route can be described using following characteristics: start time, location (set of route points), duration, length, transportation mode, novelty and attractiveness. Transportation mode in our system is one of the following: walking, running, cycling, or using motor vehicle. Detailed explanation on how we detect the transportation mode can be found in [11]. The route database in MOPSI contains 7.576 routes with 4.819.423 of individual points (6.8.2012).

C. *Recommendation methods*

In our recommendation system, we give personalized recommendations by combining various paradigms of recommendation systems. We combine collaborative filtering with information about user profile and context. As a source of recommendation we use the three different user generated data collections described above.

The challenge is how to select the most relevant items to users. First we define the context for each recommendation request. In our previous work [3] we identified four aspects of relevance: location, content, time and network. Location is physical location of the user represented by geographical coordinates (latitude and longitude). Content is determined currently based on the description of the photos, by the keywords attached with the services or by the area covered by routes. Time is considered only for routes and photos and measures the age of the item and the season of the year when item was collected. One way we utilize the social network are the ratings given by other users to services and photos. For routes we consider that route novelty and attractiveness of the destination are concepts which use the network aspect. The use of the network aspect of relevance constitutes an integral part of the system based on collaborative filtering.

Bearing these contexts in mind, we create profile for each user of MOPSI. The user profile contains user behavioral data, such as location and previous usage of service, i.e. how user interacted with the system. Namely, we store data about location users have visited and searches they have performed (location and keyword used in previous search requests).

D. *System implementation*

In this section, we describe in details how we implement the system. Brief summary of the algorithm is given. The system architecture is presented in Fig. 4.

Users access our recommendation system using MOPSI mobile application available for Symbian, Android, iOS and Windows Phone operating systems. The same application is used for generating the data collections. Alternatively, users can use the website to get recommendations. Both website and mobile applications communicate with recommendation synchronizer as shown in Fig. 4.

The recommendation synchronizer is responsible for preparing recommendation results and keeping them up to date. The recommendation results for each user are stored in a file which contains also location and time what recommendation was calculated at. Whenever the user changes position, a request is sent to the recommendation synchronizer. Moreover, if the user is less than 3 clicks away from recommendation button the same request is sent. In practice it means that the synchronizer is called whenever the user enters main screen or screen with the recommendation button. The synchronizer decides whether to recalculate recommendations or not. The recommendation is recalculated whenever user position changes significantly or available recommendation results are outdated.

The recommender (see Fig. 4) contains implementation of the recommendation algorithm. The algorithm consists of three major steps. Firstly, the items available in databases are filtered by location. Secondly, the items that were left after filtering are scored using criteria derived from the aspects of relevance. Thirdly, the scores of all three different types of items are merged together, sorted in descending order and first 20 items with the highest scores are returned as the recommendation results.

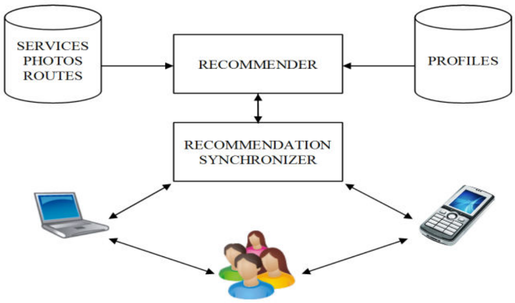


Figure 4: Architecture of the system.

IV. RECOMMENDER

In this section, we present in details the recommendation algorithm. The main conceptual steps of our algorithm are briefly described in previous section.

The algorithm input has three parameters. The first parameter identifies the user to whom recommendation is personalized for. The second parameter is the location of the user. The third parameter is the time of the recommendation request.

Firstly, we process each type of recommended items. We retrieve items from database. The full list of available items is then filtered by location to limit number of items for scoring. The remaining items are scored using several criteria based on aspects of relevance described earlier. These criteria are presented in details later in this section. The process of selection and scoring is performed for services, photos and routes separately.

Secondly, we merge all items retrieved in the previous step and sort them by total score in descending order. The recommendation results are the 20 top items in the list.

In the following subsections we describe in details how the three types of items are scored.

A. Services

Services are scored using search history, distance and rating criteria. We use both search history of all users as well as search history of the user who requested recommendation. The general search history is used to check what searches were performed in nearby locations and find what is in the area. If service keywords can be found among the keywords searched nearby, then the service is promoted by giving extra points. Furthermore, extra points are given for services with high frequency and keywords searched recently. Search history of the user is utilized in the same way.

Total score of search history consists of the following components:

$$S_H = S_{GN} + S_{GS} + S_{GF} + S_{UN} + S_{US} + S_{UF} \qquad (1)$$

where S_{GN}, S_{GS} and S_{GF} are the raw counts for keyword matches in nearby locations, within recent time and frequency of keywords in general search history and S_{UN}, S_{US} and S_{UF} are the same type of raw counts, but related to user history.

For location score we calculate distance from user to each recommendation item. By use of distance, we introduce location relevance aspect to the system.

Users can rate services through web and mobile interface of MOPSI. Services are rated by users in scale of 0 to 5. Rating and search history scores introduce content, social network and time aspects of relevance.

B. Photos

Photos are scored using search history, location, rating and time. Search history and location are used the same way as in case of services. The only difference is that service keywords are replaced by words used in photos descriptions. Rating of photos is cumulative, using a thumbs up/thumbs down system, for example a photo liked by 5 users and disliked by 2 has a rating of 3. The total score represents the rating score.

Additional score is given for recency, because the relevance of a photo decreases with time, as the places or views captured by users may change over time. Moreover, the season when the photo is taken is important for the relevance of photo, as for example winter activities are less relevant during summer.

More recent photos in the user collection are considered more relevant than old ones and the newer the photo is, the higher score it receives. Additional difference is that the score is also influenced by time of the year when the photo was taken.

Total score based on time (S_T) is for each photo calculated as follows:

$$S_T = S_A + S_Y \qquad (2)$$

where S_A is the recency and S_Y is the score for season of the year when photo was taken.

Moreover, photos are clustered into location clusters based on distances between them. Distance between photos that create a cluster is automatically decided based on distances between all photos selected for scoring. From such clusters we select only photos with highest scores. In this way we avoid recommending too many photos from the same location because we assume that the photos in same location present the same object.

C. Routes

Routes are scored using location, time and attractiveness. They are selected for scoring based on their starting point and its proximity to user's location. Main objective is to suggest user next places to visit. Therefore, only routes longer than 1 km are considered.

Location score of a route is the distance to its starting point from user location. Time score is the same as for photos.

A route is considered attractive if there is extensive collection of photos, services and other routes near its destination (destination attractiveness). Number of photos along a route increases its attractiveness (route popularity). Attractiveness score uses content aspect of relevance. Total route attractiveness score S_A is calculated using the following formula:

$$S_A = S_D + S_R \tag{3}$$

where S_D is destination attractiveness score and S_R is popularity of the route.

Similarly as in photos, clustering is used for routes. In this case we build location clusters from end points of routes. From each cluster, we select route with the highest score.

D. *Total score*

All the above scores are normalized to the scale [0..1] using the following formula:

$$N = \frac{S - MIN(S)}{MAX(S) - MIN(S)} \tag{4}$$

where S is the raw score, N is the normalized score, MIN(S) and MAX(S) are the minimum and maximum scores for each of the criterion respectively.

Final score of each service is then calculated using the following formula:

$$S_{SERVICE} = N_H + 2N_L + N_R + 1 \tag{5}$$

where N_H stands for the normalized score for search history, N_L for location, and N_R for rating. Instead of using time relevance, a constant of one point is added in order to promote services for recommendation, because they are assumed to originate from a trusted source and therefore more relevant than older photos from user collection. The location score is multiplied by two to emphasize the importance of the location.

Final score of each photo item is calculated in the same way as services, having an additional time score:

$$S_{PHOTO} = N_H + 2N_L + N_R + N_T \tag{6}$$

where N_T stands for time.

Final score of each route is calculated using the following formula:

$$S_{ROUTE} = 2N_L + N_A + N_T \tag{7}$$

where N_A stands for attractiveness score.

V. SYSTEM EVALUATION

As we described in [12], the evaluation of user satisfaction is an important part of the evaluation of recommendation system. Moreover, as stated there, the feedback from users can be used to improve the system performance. The current improvements of our system are based on previously conducted experiments and on collected feedback. For example, recommending too many photos from one location was pointed out as a drawback of our previous system.

For testing our current system we used the same qualitative experimental settings as in previous work. We chose the city of Joensuu as the location of our tests, because the biggest number of service users has data collected in the area. Within Joensuu borders we selected several locations of various types. The locations list included city center, living areas, industrial area and recreational areas. We checked whether the recommendation is relevant for users. We also evaluated the items that were scored, but not recommended, in order to find out what relevant results may have been overlooked by the implemented scoring function.

Recommendations in city center always give many cafeterias, pizzerias and bars taken from service database. In addition, the system provides additional recommendations such as sport places and shops taken from photo collection. Also services that are not in service database, but are in photo collection of users, are recommended. Experiments have shown that all factors have impact on the recommendation results. For example, in suburban area with many housing blocks of flats and services nearby, there are many eating places recommended, but they are chosen based on rating and search history. Same example shows well that our recommendation system chooses relevant photos from the collection, such as shop, kiosk and mailbox, whilst general photos of streets, houses and people are skipped, although located nearby. In suburban are, on the other hand, where user generated collections are smaller, location has bigger impact on the relevance score.

In the course of experiments with our system, we noticed that it gives results expected by users mainly in case of services such as restaurants, bars and outdoor activities places (skiing, swimming). It does not perform as well in case of shops though, because of lack of data in users collections. There are cases where recommendation system does not suggest relevant only because it lacks description input by the user.

Experiments indicate that including time score is very important. The example in Fig. 5 shows recommendation in the same location in the middle of lake close to group of islands both in winter and in summertime. The place is popular destination for cross country skiers in winter. In summer the place is possible to access by own boat only and therefore rarely visited. Our system recommends the islands as place to visit when the lake is frozen and skiing tracks exist, because there are many routes recorded by users in wintertime. On the other hand, in summer it recommends barbecue places and swimming places on lakeside in town as no user activity is visible close to the islands.

Fig. 6 demonstrates the use of the clustering for photo recommendation. On the left of the figure we see photos recommended when location clustering is not applied. There are several photos located in a very small perimeter. Most of the photos are inside or around the same building. On the right side of the figure, only one photo is chosen to represent the cluster formed. Moreover, our system selects photos with best scores from cluster. In this way, we avoid recommending irrelevant test photos described above that has high distance score, but theirs other scores are low.

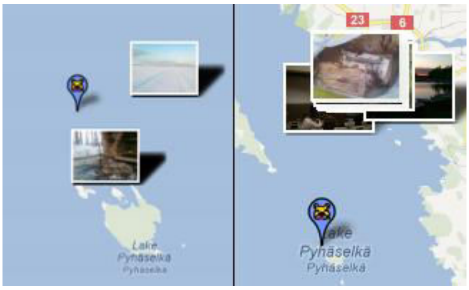


Figure 5: Recommended photos in winter (left) and summer (right).

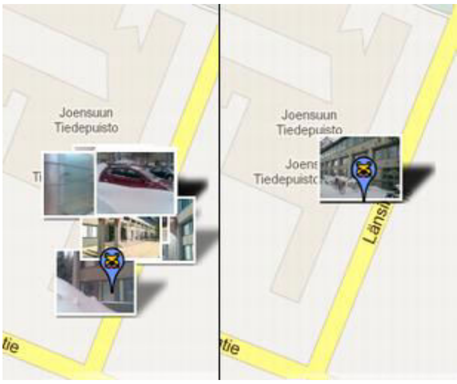


Figure 6: Photo recommendation without location clusters (left) and with clustering applied (right).

VI. CONCLUSIONS AND FUTURE WORK

We have designed a context aware personalized recommendation system. Database of recommended items has free form and is generated by the users of MOPSI without any data cleansing. In this paper we study how to mine knowledge from user generated collections. We recommend three types of items: services, photos and routes. The goal of the system is to recommend points of interests to visit in user's surrounding.

The conducted experiments demonstrate that our system selects relevant items to recommend. Changes of algorithm we proposed in comparison to previous version of the system are beneficial for recommendation result quality.

However, despite the fact of positive feedback from system users, there is room for further improvements. Recommending routes was introduced to the recommendation system recently and scoring criteria should be improved. In some tests relevant routes were missed, because the routes had lower score than other items so they were not selected for recommendation. Moreover, routes recommendation brings new challenges. Route processing and computing the attractiveness score is time consuming. Therefore, we consider storing route statistics in database in a similar way as users' profiles are stored.

The clustering concept can be expanded for the photo collection. We can create not only location clusters, but also content clusters based on photo descriptions. The photo content can be analyzed automatically and such keywords assigned in the process could be stored. Such solution will prevent us from relying merely on user provided description of photo as descriptions are often missing in photo collections.

Furthermore, the weights for different elements of total score could be adjusted automatically.

REFERENCES

[1] Bellotti, V. et al., "Acitivity-based serendipitous recommendation with the Magitti mobile leisure guide", ACM SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy, April 2008.

[2] Clements, M., Serdyukov, P., de Vries, A. P., and Reinders, M. J. T., "Personalised travel recommendation based on location co-occurrence", The Computing Research Repository, 2011.

[3] Fränti, P., Chen, J., and Tabarcea, A., "Four aspects of relevance in sharing location-based media: content, time, location and network", Int. Conf. on Web Information Systems and Technologies, Noordwijkerhout, The Netherlands, May 2011.

[4] Kim, Y., and Cho, S.-B., "A recommendation agent for mobile phone users using Bayesian behavior prediction", Int. Conf. on Mobile Ubiquitous Computing, Sliema, Malta, October 2009.

[5] Levandoski, J.J., Sarwat, M., Eldawy, A., and Mokbel, M.F., "LARS: a location-aware recommender system", Int. Conf. on Data Engineering, Washington D.C., USA, April 2012.

[6] Li, L.H., Lee, F.M. and Chen, Y.C, "A multi-stage collaborative filtering approach for mobile recommendation", Int. Conf. on Ubiquitous Information Management and Communication, Suwon, Korea, January 2009.

[7] Savage, N.S., Baranski, M., Chavez, N.E., and Höllerel, T., "I'm feeling LoCo: A Location Based Context Aware Recommendation System", Lecture Notes in Geoinformation and Cartography, 2012.

[8] Schilke S.W., Bleimann, U., Furnell, S.M., Phippen, A.D., "Mand interest-based recommendation", Internet Research, Vol. 14, Iss: 5, pp. 379 – 385, 2004.

[9] Takeuchi, Y., and Sugimoto, M. "CityVoyager: An Outdoor. Recommendation System Based on User Location History", Int. Conf. on Ubiquitous Intelligence and Computing, China, 2006.

[10] Tuan, C.C., Hung, C.F., and Kuei, T.C., "Location dependent collaborative filtering recommendation system", Int. Conf. on Future Network Technologies, Qingdao, China, August 2011.

[11] Waga, K., Tabarcea, A., Chen, M., and Fränti, P., "Detecting movement type by route segmentation and classification", in press.

[12] Waga, K., Tabarcea, A., and Fränti, P., "Context-aware recommendation of location-based data", Int. Conf. on System Theory, Control, and Computing, Sinaia, Romania, October 2011.

[13] Yang, F., and Wang, Z.M., "A mobile location-based information recommendation system based on GPS and WEB2.0 services", WSEAS Transactions on Computers, Vol. 8 Iss: 4, pp. 725 – 734, April 2009.

[14] Yoon, H., Zheng, Y., Xie, X., and Woo, W., "Social itinerary recommendation from user-generated digital trails", Personal and Ubiquitous Computing, June 2011.

[15] Zheng, V. W., Zheng, Y., Xie, X., and Yang, Q., "Learning from GPS data for mobile recommendation", Artificial Intelligence Journal, February 2012.

[16] Zheng, Y., Zhang, L., Ma, Z., Xie, X., and Ma, W.-Y., "Recommending friends and locations based on individual location history", ACM Transactions on the Web, Vol. 5, No. 1, February 2011.

Paper IV

P. Fränti, K. Waga and C. Khurana,
"Can social network be used for location-aware recommendation?"
11th Int. Conf. on Web Information Systems & Technologies
(WEBIST'15),
558 – 565, Lisbon, Portugal, 2015.

Can Social Network Be Used for Location-aware Recommendation?

Pasi Fränti, Karol Waga and Chaitanya Khurana
School of Computing, University of Eastern Finland, Joensuu, Finland
{franti, kwaga, chaitkh}@cs.uef.fi

Keywords: Social Network, Location-Aware Search, Recommendations, Personalization.

Abstract: Our goal is to give recommendations for mobile users about interesting places around his current location. The only input is the user, location and time. In this work, we study whether the social network of the user can be utilized for improving recommendations. We will answer the following two questions: (1) can we measure user similarity based on their Facebook profile and location history, (2) do these imply usefulness for the recommendations.

1 INTRODUCTION

Location-based services have become widely used due to the fast development of positioning systems in multimedia phones. We study recommendation system for a mobile user who wants information about nearby services such as shops and restaurants. User can make a query specified by keyword(s), or he can just ask general recommendation without any keywords as input (see Fig. 1). In the latter case, the relevance of a service must be determined merely by other factors such as user location, time and personal preferences. In (Fränti et al., 2011), relevance of a recommendation was considered to consist of four aspects:

- Location
- Time
- Content
- User and his/her network

Location is the key aspect but not the only one, see Fig. 2. In (Waga et al., 2012), recommendations were influenced by the overall search history by giving higher rating for entries with popular keywords in their title or tags, see Fig. 3. Extra points were given to keywords that were used often, used recently, or search in the nearby location of the user. Keywords used by the user himself were also given higher score. Recommended items were taken both from Mopsi service directory, and from the photo collections of the users.

In this work, we study whether a network of the user can be used for improving recommendation. Social knowledge was explored in (Bao et al., 2012) by considering opinions of local experts in the given area. This can be useful for improving rating of the services by utilizing users whose opinions matter most. User network can also become useful when making recommendations, especially for the

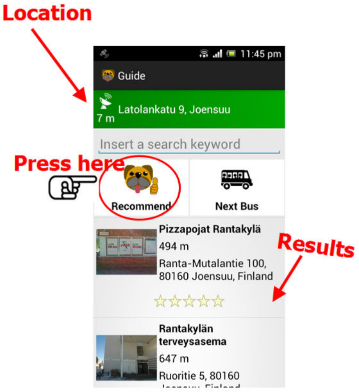


Figure 1: Recommendation in Mopsi (http://cs.uef.fi/mopsi).

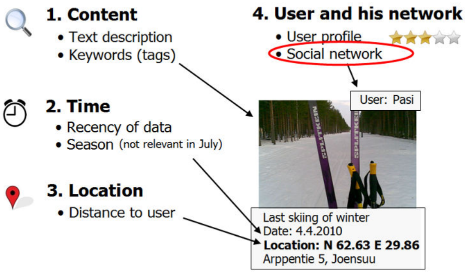


Figure 2: Four aspect of relevance in geo-tagged photo.

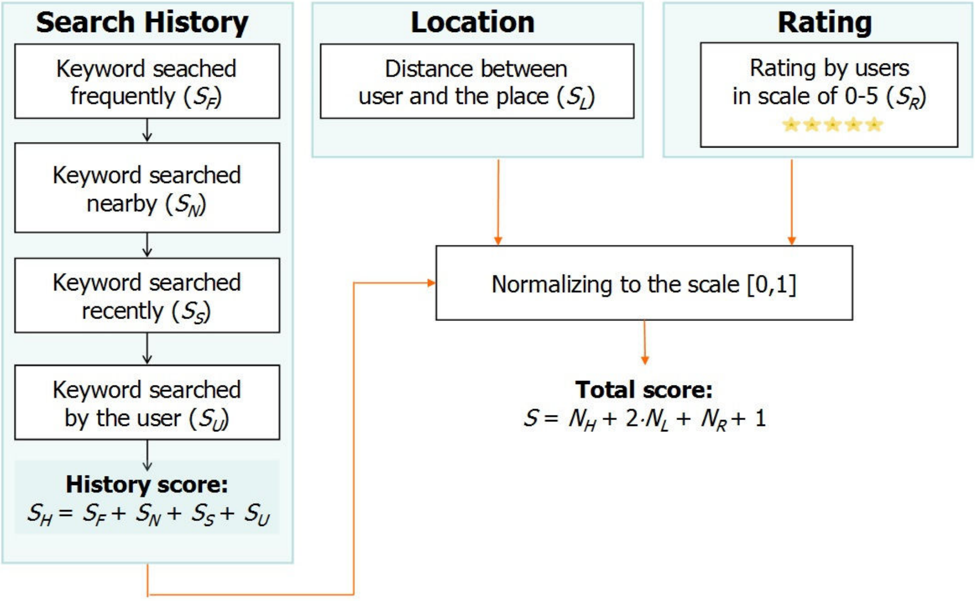

Search History
Keyword seached frequently (S_F)
Keyword searched nearby (S_N)
Keyword searched recently (S_S)
Keyword searched by the user (S_U)
History score:
$S_H = S_F + S_N + S_S + S_U$
Location
Distance between user and the place (S_L)
Rating
Rating by users in scale of 0-5 (S_R)
Normalizing to the scale [0,1]
Total score:
$S = N_H + 2 \cdot N_L + N_R + 1$

Figure 3: Scoring recommendations based on relevance to user.

so-called *cold start users*, from whom we have very little or no previous history data. Profiles and parameters used for their friends and similar users can provide good initial guess for personalizing the recommendations (Yang et al., 2012).

For utilizing the network, it is not clear what type of network should be used, and how much a given user should influence the recommendation for another user. For this task, we study how similar two users are when measured by the following features:

1. Friendship in Facebook
2. Pages liked in Facebook
3. Places visited in Mopsi

We perform qualitative experiment with a small set of nine Mopsi users. We study the facebook pages the users like, and the frequency of the places they have visited in Joensuu. We study how much the user similarity according to these features correlate to the subjective opinions of the user themselves, and also how they useful they rank the recommendations of the other users in the location-aware recommendation context.

The main findings are that the user similarity correlates with all the features studied but not very strongly. There is mild correlation with the user locations (0.28) and the pages liked (0.47) but the strongest correlation is with the facebook friendship. In most cases, users ranked their facebook friends as more similar than the others. However, when asked how useful they would expect the data (photos in Mopsi) of the other users, all the correlations decreases and indicate that these features are not easy to utilize on location-aware recommendation system.

2 UTILIZING USER NETWORKS

So far, user networks have been the least utilized aspect in Mopsi recommendations. The service is public to entire world and there are no friend-to-friend connections. Currently the only user network implemented is the one suggested by clustering the users according to their location, see Fig. 4. This can be used to inform people who else is in the same area. We next discuss possible types of network from the following perspectives:

- Social network vs. information sharing network
- Buddy network vs. stranger network
- Selected friends vs. automatic ad hoc network
- On-line vs. offline network

For a more extensive taxonomy of social web sites, see (Kima et al., 2010).

2.1 Effectiveness of the Network

By far the most widely used networks nowadays are the social networks implemented by Facebook, Twitter, Google+, Instagram and other similar platforms where users explicitly specify with whom they share their data. Social network has indeed very strong influence whose data is more relevant to the user but it is not the only possible network.

Users in general are more interested what their friends are doing than other people in general. However, in recommendation system, the relevance of the information is more important than the social aspect. In location-aware recommendation system, users are seeking for information around his current location. A user visiting the place often is therefore more likely to have more relevant information than a friend. In this view, we have more pragmatically driven information sharing platform rather than merely a social network.

Another aspect of social network is that how well the people connected actually know each other. According to the *small-world* phenomenon (Watts and Strogatz, 1998), we can reach anyone in the world by six steps, on average.

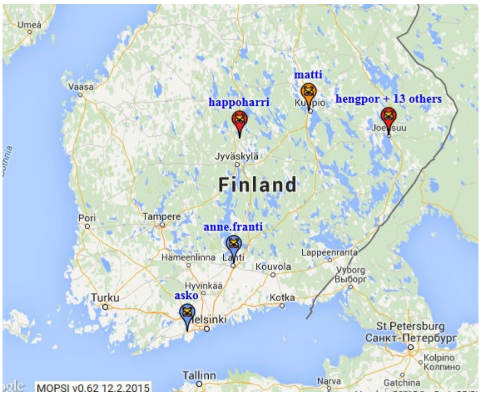


Figure 4: Example of clustering users according to their location.

It was shown in (Barrat and Weigt, 2000) that even a small amount of disorder (randomness) in the network is able to trigger the small-world behaviour even if the network was otherwise strongly clustered. Therefore, the connectivity of the network is not the bottleneck but the quality of the links is.

Network like Facebook is not really friend network, but a term like *buddy network* would be more appropriate. Due to social pressure, people often try to be as connected as possible, which does not really make sense from the efficiency point of view. Having 400 Facebook friends does not imply that the person has 400 real friends; a more likely number would be about 10 or less. Nevertheless, the people who are linked together know each other, and the small-world phenomenon applies.

From information distribution point of view, the relevance of the information sent via network is affected by how many people we are connected to, and how often we use these links. Instead of sharing information via a large number of links, few strong connections are likely to be more effective than a large number of weaker links. The strength of the connection is therefore more important than the connection itself.

Contrary to social networks, strangers may also be linked together because of sharing the same interest. In *cough surfing*, people offer accommodation to others without financial compensation (Bolici, 2009). The key aspects in such *stranger network* are the reputation and trust between the users. In Mopsi, only information is traded but in the same way, the reputation of the author influences how trustworthy we consider his/her data. Recommendations can be used to build up the trust, and improve the quality of the information.

2.2 Automatically Created Networks

For computer scientists, anything that can be automated is always worth to consider. Users can be linked based on their behaviour how they use the service (Gratz and Botev, 2009), or simply according to their location. In Mopsi, the location is taken into account in the recommendation system already, but the similarity of the users is not yet utilized. In (Bacon and Dewan, 2009), similar users are recommended to each other. Once there will be enough users in the service, similarity can be used to offer personalized search result.

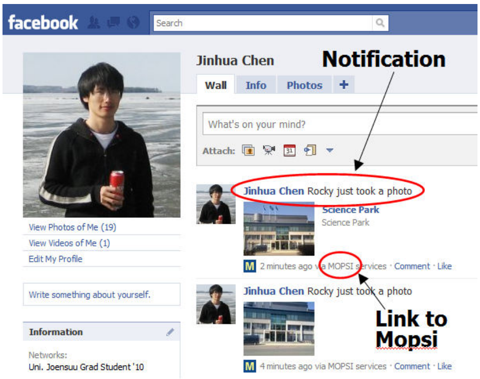
A more ambitious ad hoc network is considered in (Wu et al., 2009) using face analysis technique to identify people in photos, and use this information to create more complex social network automatically. If more thorough content analysis could be successfully done, people with the same hobbies could be connected automatically.

Another approach is to combine location-based service and social network from two independent components as done in (Simon et al., 2009). One can then focus on developing the location-based media collection and service directory, and utilize existing network for user identity and all the social networking functionalities that come along. In Mopsi, we implemented login using Facebook credentials, which allows users to share their Mopsi data in Facebook: the system generates (optional) status update to inform their network buddies as shown in Fig. 5. Data is still stored also in Mopsi but all the discussion happens in Facebook.

2.3 Behaviour in a Public Network

The nature of being an on-line or offline network affects how people use it. In our case, the data collection itself has online nature but since there is no online conversation forum in Mopsi, the system is more like offline by its nature.

Personality also affects how people use social networks. Extravert personalities are more likely to engage social activities but according to (Ross et al., 2009), personality has much smaller effect than expected on how they use Facebook. For example, social person is likely to join more groups but it does not reflect much on the size of the network, or how extensively the communicative functions are used. This can be partly explained by the fact that Facebook is less widely used for on-line chatting than other forums for live communication.



Jinhua Chen Rocky just took a photo

Science Park
Science Park

2 minutes ago via MOPSI services

Jinhua Chen Rocky just took a photo

4 minutes ago via MOPSI services

Figure 5: Facebook status update via Mopsi photo upload.

The level of neuroticism in personality, however, affects on how much people preferred text (writing on the wall) or sharing photos in Facebook. People with higher sensitivity to threat use more textual expression and less photo sharing because it was more controllable due to its off-line nature Ross et al., 2009). Another study showed that the identity people present in their social network can differ a lot from their real personalities. It was observed that the image people gave was more real in off-line chatting environment than in offline social network (Zhao et al., 2008).

The privacy issue can also be important for people who would want to use the service, but wish not to reveal their identity or even location. Methods have been developed specifically to prevent the system to combine user's identity and location (Takabi et al., 2009), which actually contradicts our goals of specifically sharing the user location. This reflects the privacy concern, which the social network and information sharing evidently weaken if not adequately solved.

In Mopsi, the motivation is to encourage people to share their information via their personal collection, and use their network for the same. We should encourage people to share as much information as possible so that it would have high coverage, but on the other hand, keep the quality of the information trustworthy so that it would be relevant and therefore useful to recommend to others. Division of the service to two different concepts – personal collection and service database – aims at reaching both of the goals at the same time. How to transfer data from the personal collection to the open database is a point of further development.

3 SIMILARITY OF USERS

We study next empirically the connections between users in Facebook and in Mopsi. We selected nine volunteers who work either in our lab or nearby (see Table 1). They all live in Joensuu use both Mopsi and Facebook, and know each other at some level. Most of them are linked in Facebook as well. We asked them to evaluate their relationship and rank the other people from 1 to 8 using the following two criteria:

Q1: *How similar you find the person is to you?*
Q2: *How useful you find his/her Mopsi photos?*

For the second question, context is that does he recommend, via his/her Mopsi postings, useful and interesting places to visit in future. The first question was to measure similarity whereas the second tries to explore whether the usefulness goes beyond similarity and friendship. The resulting rankings are shown in Tables 2 and 3. Pink background of a cell is used to indicate that the users are not linked in Facebook. As expected, if one considers the other similar, they are also connected in Facebook. In this regard, similarity and connection in social network seems to correlate.

3.1 Analysis of the User Evaluations

Detailed inspection of the data reveals that the similarity ranking is quite subjective. The sum values show that certain people tend to be more often “similar” than others. For example, Radu, Pasi and Andrei have average rankings of 1.5, 2.8 and 3.0. In specific, Radu is the most similar for five other users, and ranked 2nd or 3rd for the rest. By common sense, everyone cannot be just like Radu, but knowing him we conclude that most people would not mind being like him. Further analysis of the FB data (not included here) shows that the more the photos and status updates of a particular user are liked and commented, the more similar he/she is considered.

The two rankings have reasonable high correlation with each other (0.52) but there are few differences. In the usefulness evaluation Pasi becomes the highest ranked due to frequent publishing of travel photos. Also Julinka’s photos are considered more useful for the same reason. Otherwise, the usefulness and similarity rankings are quite similar. However, we asked how useful users *expect* the data of their friends to be, but in fact, the expectation may not match the reality. Some low rankings might be biased towards low publication activity rather than the usefulness of these photos.

Table 1: Volunteers participating in the experiment.

	Mopsi			Facebook	
	photos	places	visits	friends	pages
Andrei	676	96	676	463	285
Julinka	3850	122	2116	229	154
Mikko	190	84	292	55	14
Oili	6467	164	1261	298	63
Pasi	9716	208	3847	88	67
Radu	1417	122	912	298	19
Rezaei	716	85	587	193	16
Chait	63	22	53	580	195
Jukka	991	126	682	142	120

Table 2: User similarity based on their own view.

	Andrei	Julinka	Mikko	Oili	Pasi	Radu	Rezaei	Chait	Jukka
Andrei	-	7	8	4	2	1	3	6	5
Julinka	2	-	4	3	6	1	5	7	8
Mikko	7	8	-	5	1	2	4	6	3
Oili	3	5	7	-	2	1	4	8	6
Pasi	3	8	5	4	-	2	6	7	1
Radu	1	8	4	5	2	-	3	7	6
Rezaei	4	7	2	6	1	3	-	8	5
Chait	2	8	4	7	5	1	3	-	6
Jukka	2	7	5	4	3	1	8	6	-
Average:	**3.0**	**7.3**	**4.9**	**4.8**	**2.8**	**1.5**	**4.5**	**6.9**	**5.0**

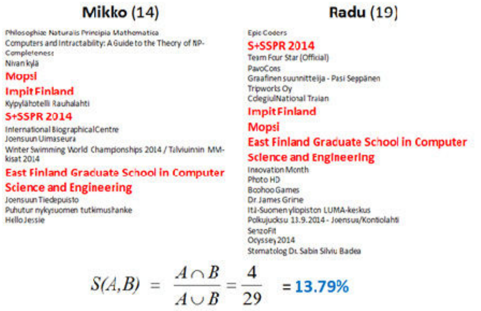
Table 3: Expected usefulness of friend’s photos.

	Andrei	Julinka	Mikko	Oili	Pasi	Radu	Rezaei	Chait	Jukka
Andrei	-	5	8	4	1	2	6	7	3
Julinka	2	-	6	3	4	1	5	7	8
Mikko	4	1	-	8	2	6	7	5	3
Oili	4	5	7	-	1	2	6	8	3
Pasi	2	7	1	4	-	5	8	6	3
Radu	2	5	7	4	1	-	6	8	3
Rezaei	6	2	7	3	1	5	-	8	4
Chait	3	7	8	4	2	1	6	-	5
Jukka	3	6	5	4	1	2	8	7	-
Average:	**3.3**	**4.8**	**6.1**	**4.2**	**1.6**	**3.0**	**6.5**	**7.0**	**4.0**

3.2 Similarity in Page Liking

For testing the similarity of users, we compared how many same Facebook pages the users liked. For example, Mikko and Radu like four same pages (*Mopsi, Impit Finland, S+SSPR 2014 and East Finland Graduate School of Computer Science & Engineering*), out of total 29 pages that either both or one of them likes. Using these numbers, we define their similarity by Jaccard coefficient as the number of matches divided by the total number of pages: 4/29 = 14%, see Figure 6.

The similarity values for the page likes are shown in Table 3. As expected, lowest values are typically among users who are not linked in Facebook. The page liking correlates also reasonably well (0.47) with the user similarity values (Table 2) but the correlation with the usefulness values (Table 3) is much smaller (0.17). Therefore, even if user similarity could be estimated by their user profiles in facebook, using it for location-aware recommendation would still be questionable.



Mikko (14)

Philosophiae Naturalis Principia Mathematica
Computers and Intractability: A Guide to the Theory of NP-Completeness
Nivan kylä
Mopsi
Impit Finland
Kylpylähotelli Rauhalahti
S+SSPR 2014
International Biographical Centre
Joensuun Uimaseura
Winter Swimming World Championships 2014 / Talviuinnin MM-kisat 2014
East Finland Graduate School in Computer Science and Engineering
Joensuun Tiedepuisto
Puhutun nykysuomen tutkimushanke
Hello Jessie

Radu (19)

Epic Coders
S+SSPR 2014
Team Four Star (Official)
PavoCons
Graafinen suunnittelija - Pasi Seppänen
Tripworks Oy
ColegiulNational Traian
Impit Finland
Mopsi
East Finland Graduate School in Computer Science and Engineering
Inovation Month
Photo HD
Boohoo Games
Dr James Grime
Itä-Suomen yliopiston LUMA-keskus
Polkujuoksu 13.9.2014 - Joensuu/Kontiolahti
SenzoFit
Odyssey 2014
Stomatolog Dr. Sabin Silviu Badea

$$S(A,B) = \frac{A \cap B}{A \cup B} = \frac{4}{29} = 13.79\%$$

Figure 6: Sample similarity calculations of users based on their likes in Facebook.

Table 4: Similarity values for Facebook page likings (%).

	A	J	M	O	P	Ra	Re	C	JP
Andrei	-	3	2	3	5	2	2	3	2
Julinka	3	-	1	2	1	1	1	1	1
Mikko	2	1	-	7	6	25	16	3	5
Oili	3	2	7	-	8	6	6	3	4
Pasi	5	1	6	8	-	6	4	2	4
Radu	2	1	25	6	6	-	14	3	5
Rezaei	2	1	16	6	4	14	-	2	3
Chait	3	1	3	3	2	3	2	-	1
Jukka	2	1	5	4	4	5	3	1	-

Another issue is that liking exactly the same page is not likely to happen in larger scale. For example, if one person likes *McDonalds* and the other one a local brand *Hesburger*, they are still similar as they like fast food restaurants. We considered counting matches of the categories the pages belong to. Facebook has roughly 54 million pages, which all belong to 107 predefined categories. For example, McDonalds and Hesburger are both in *fast food* category. The same Jaccard measure can still be applied.

However, results using category matches show even lower correlation because the categories are too general. We therefore dropped this idea and use page liking as such. Fig. 7 shows part of the similarity graph for the set of test users.

3.3 Similarity in Location History

For studying location activity, we selected 293 places from Mopsi services as the visit places in Joensuu. We recorded user activities until 31.12.2014 as follows: (1) places where they took photos, (2) places where tracking a route was started or ended. Each activity is counted as a visit to the nearest place to the location of the activity. We used only locations within the bounding box (28.65E, 63.44N, 31.58E, 62.25N) that roughly covers Joensuu city and the rural areas of the municipality. There are 10,426 visits in total. The number of visits of each user is reported in Table 1.

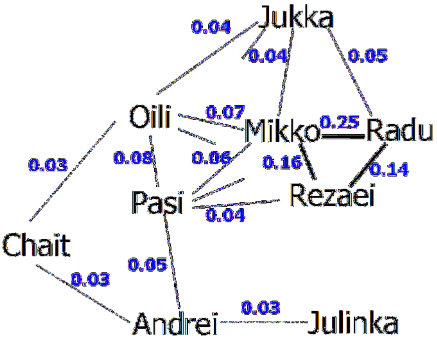
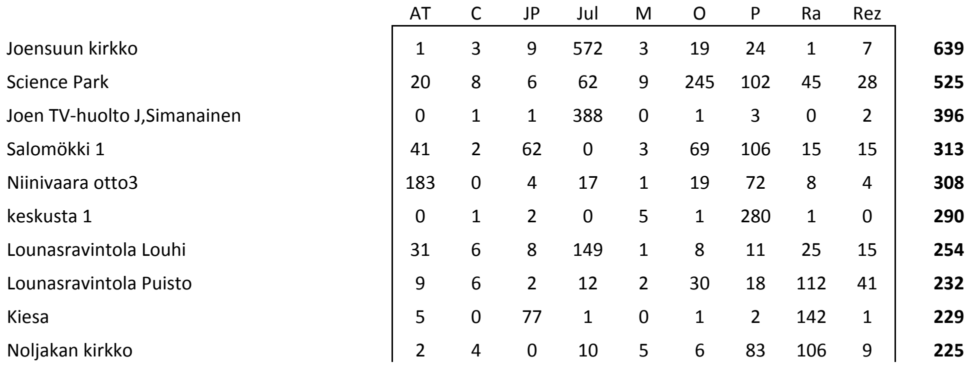


Figure 7: Similarity graph constructed from the biggest similarities in page likings.

The location data of a user forms a frequency histogram consisting of 293 bins. The most popular places with the corresponding visit frequencies are listed in Fig. 8.

Location similarity of two users i and j are calculated using Bhattacharyya distance between their histograms:

$$D_B = -\ln \sum \sqrt{p_i \cdot p_j}$$



	AT	C	JP	Jul	M	O	P	Ra	Rez	
Joensuun kirkko	1	3	9	572	3	19	24	1	7	**639**
Science Park	20	8	6	62	9	245	102	45	28	**525**
Joen TV-huolto J,Simanainen	0	1	1	388	0	1	3	0	2	**396**
Salomökki 1	41	2	62	0	3	69	106	15	15	**313**
Niinivaara otto3	183	0	4	17	1	19	72	8	4	**308**
keskusta 1	0	1	2	0	5	1	280	1	0	**290**
Lounasravintola Louhi	31	6	8	149	1	8	11	25	15	**254**
Lounasravintola Puisto	9	6	2	12	2	30	18	112	41	**232**
Kiesa	5	0	77	1	0	1	2	142	1	**229**
Noljakan kirkko	2	4	0	10	5	6	83	106	9	**225**

Figure 8: Most popular places and their corresponding visit frequencies.

Table 5: Location similarities.

	A	J	M	O	P	Ra	Re	C	JP
Andrei	-	0,33	0,32	0,34	0,54	0,50	0,51	0,38	0,45
Julinka	0,33	-	0,29	0,45	0,52	0,40	0,40	0,46	0,35
Mikko	0,32	0,29	-	0,27	0,53	0,59	0,38	0,30	0,37
Oili	0,34	0,45	0,27	-	0,46	0,37	0,51	0,60	0,30
Pasi	0,54	0,52	0,53	0,46	-	0,68	0,68	0,52	0,54
Radu	0,50	0,40	0,59	0,37	0,68	-	0,58	0,45	0,65
Rezaei	0,51	0,40	0,38	0,51	0,68	0,58	-	0,53	0,56
Chait	0,38	0,46	0,30	0,60	0,52	0,45	0,53	-	0,42
Jukka	0,45	0,35	0,37	0,30	0,54	0,65	0,56	0,42	-

where the summation is done over all the 293 entries, and p_i, p_j are the relative frequencies of the given place. For example, Andrei has frequency 183/676 = 0.19 for the Niinivaara Otto 3, which is an ATM machine near to his home. Other similar visits happens near the users' homes (Julinka used to live opposite to Joensuu kirkko), or working place (everyone except JP works in Science Park).

The similarity results are summarized in Table 4. Only mild correlation (0.28) is recognized with the similarity of the users based on their personal views and their location history, and even smaller with the usefulness measure (0.17). Open question is how much the choice of the methodology influences the results, and if some choices made there could be changed. For example, the number of places and how they are chosen. High frequencies of the home and work places of the users had also a relative large effect: not living or visiting the same area might significantly decrease the similarity of such user.

Nevertheless, the results indicate that the location history has relatively small impact on user similarity and it is not clear how they could be used on improving recommendations.

4 CONCLUSIONS

Small-scale study was made with nine Mopsi and Facebook users to find out whether user similarity and their expected usefulness for recommendation could be predicted from Facebook profile and location history. Based on the results we observed that matching page likes in Facebook correlated with user similarity whereas the location history had only mild correlation. Neither of these statistics predicts which user's data is expected to be most useful.

However, we also noticed that if a user gives many likes and comments of the photos of another user, then he considers this user more similar than others; and what's more important, consider his data more useful for location-aware recommendation. We therefore conclude that, yes, social network can be used for improving recommendations, but not with the data (page likes and location history) in the way studied in this work.

Nevertheless, the results showed correlations and revealed potentially useful factors indicating user similarity. These findings should be confirmed by large-scale testing. We also plan to make similar study using likes and comments, which have been applied for recommending events and friends in (De Pessemier et al, 2013).

REFERENCES

J. Bao, Y. Zheng, M.F. Mokbel, "Location-based and preference-aware recommendation using sparse geo-social networking data", Int. Conf. on Advances in Geographic Information Systems (SIGSPATIAL), 199-208, Redondo Beach, CA, 2012.

K. Bacon, P. Dewan, "Towards automatic recommendation of friend lists", *CollaborateCOM*, Crystal City, Washington DC, Nov 2009.

A. Barrat and M. Weigt, "On the properties of small-world network models", *Eur.Phys.J. B*, 13, 547-560, 2000.

F. Bolici, "No hotel in D.C.", *CollaborateCOM*, Crystal City, Washington DC, Nov 2009.

T. De Pessemier, J.Minnaert, K. Vanhecke, S. Dooms, L. Martens, "Social Recommendations for Events", ACM Conf. on Recommender Systems, Hong Kong, China, October 2013.

P. Fränti, J. Chen and A. Tabarcea, "Four aspects of relevance in location-based media: content, time, location and network", *Int. Conf. on Web Information Systems & Technologies (WEBIST'11)*, Noordwijkerhout, Netherlands, 413-417, May 2011.

P. Gratz, J. Botev, "Collaborative filtering via epidemic aggregation in distributed virtual environments" *Collaborate COM*, Crystal City, Washington DC, Nov 2009.

C.J. Hutto, S. Yardi, E. Gilbert, "A longitudinal study of follow predictors on twitter",*SIGCHI Conf. on Human Factors in Computing Systems* (CHI'13), 821-830, 2013.

W. Kima, O.-R. Jeong, S.-W. Lee, "On social web sites", *Information Systems*, 35, 215-236, 2010.

C. Ross, E. S. Orr, M. Sisic, J. M. Arseneault, M. G. Simmering, R. R. Orr, "Personality and motivations associated with Facebook use", *Computers in Human Behavior*, 25, 578-586, 2009.

J.R. Simon, D.R. Gonzalez, C.F. Grande, C.E. Gomez, A.P. de la Llave, F.O. Lacalle, K.D.R. Permingeat, "NEMOS: Working towards the 'social' mobile phone", *ICME 2009*, 1784-1788, New York City, July 2009.

H. Takabi, J.B.D. Joshi, H.A. Karimi, "A collaborative k-anonymity approach for location privacy in location-based services", CollaborateCOM, Crystal City, Washington DC, Nov 2009.

K. Waga, A. Tabarcea and P. Fränti, "Recommendation of points of interest from user generated data collection", IEEE Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), Pittsburgh, USA, 2012.

D. Watts and S. Strogatz, "Collective dynamics of 'small-world' Networks", Nature, 393, 440-442, 1998.

P. Wu, W. Ding, Z. Mao, D. Tretter, "Close & closer: "Discover social relationship from photo collections", ICME 2009, 1652-1655, New York City, July 2009.

X. Yang, H. Steck, Y. Guo, Y. Liu, "On Top-k Recommendation using Social Networks", ACM Conf. on Recommender Systems, Dublin, Ireland, 67-74, Sept. 2012.

S. Zhao, S. Grasmuck, J. Martin, "Identity construction on Facebook: Digital empowerment in anchored relationships", Computers in Human Behavior, 24, 1816-1836, 2008.

Paper V

K. Waga and P. Fränti,
"Similarity of mobile moving users based on sparse location history",
manuscript, 2015.

Similarity of Mobile Users Based on Sparse Location History

Karol Waga[a] and Pasi Fränti[a]

[a]*School of Computing, University of Eastern Finland, Finland*

ABSTRACT

We propose a method to measure similarity of users based on their location history. Instead of continuous movement trajectories, we consider sparse location data originating from specific user activities. We map each activity point into the nearest location in a predefined set of fixed places. The problem is then formulated as histogram comparison. We compare several measures including L_1, L_2, L_∞, ChiSquared, Bhattacharyya and Kullback and Leibler divergence using both crisp and fuzzy histograms. Results show that all methods are suitable for the task except L_2 and L_∞.

1. Introduction

Similarity of users has been widely used in recommender systems based on the assumption that similar users are interested in the same things. Collaborative recommender systems (Adomavicius and Tuzhilin, 2005) estimate relevance of an item to a given user based on ratings given by similar users. To find similar users most of the recommendation systems searches for the common items that the users have rated (Adomavicius and Tuzhilin, 2005). This approach is used in online shops, movie databases and similar recommendation systems.

The knowledge of similar users has been applied to improve retail experience by finding correlation between buying and browsing behavior. Similarity of users have also been used for recommending events and friends in (De Pessemier et al., 2013), and provide good initial guess for personalizing the recommendations especially for new users (Yang et al., 2012). Similarity of users have also been measured based on how they tag for bookmarking purpose (Li et al., 2008a).

In (Fränti et al., 2015), we studied whether social network can be used for improving location-aware recommendations. The results showed that user's own understanding of the similarity correlates more with the similarity of their page likings and less with the similarity of their location histories, for which only minor correlation was detected. The same order of importance was observed in (Guy et al.,2010): people value most the things they have common, then the places where they are active, and least important was to know the same people.

In location-aware recommendations, however, opinions of local experts in the given area can be more valuable than just the similarity of the user (Bao et al., 2012). This can be useful for improving rating of the services by utilizing users whose opinions matter most. In general, knowing the similarity of location history can provide additional information for improving recommendation. In this paper, we study how the similarity of users can be measured from a limited amount of location data.

One approach is to analyze complete trajectories of the user movements. In (Ying et al., 2010), potential friends are recommended based on users' movement trajectories. So-called *stay cells* are created based on detected stops, which are considered important places because user stayed there longer time. Similarity of trajectories are then measured based on their longest common subsequence giving higher weight of the longer patterns. In (Liu et al., 2012), revised version of the longest common subsequence is applied by partitioning the trajectories based on speed and detected turn points. The similarity score is based on both geographic similarity and the semantic similarity.

In (Biagioni et al., 2013), similarity of a person's days is assessed based on the trajectory by discovering their semantic meaning. The data is collected from tracking users' cars and pre-processed by detecting stop points. Most common pairs of stops are assumed to be user's home and work locations. Dynamic time warping of the raw trajectories using geographic distances of the points is reported to work best. In (Wang et al., 2012), personalized search for similar trajectories is performed by taking into account user preferences of which parts of the query trajectory is more important.

Complete trajectories are not always available and the similarity must then be measured based on sparse location data such as visits, favorite places or check-ins. In (Li et al., 2008b], user data is hierarchically clustered into geographic regions. A graph is constructed from the clustered locations so that a node is a region user has visited, and an edge between two nodes represents the order of the visits to these regions. This method still relies on the order of the locations visited.

In this paper, we study how the similarity of users can be measured based on their location history when the entire GPS trajectories are not available. We use single location points originating from geo-tagged photos, and the start and end points of movements. Other activity points that could be used are stay points, i.e. the places where used stayed longer than 30 mins, as in (Zheng et al., 2010).

We propose to measure the similarity between users by taking each user activity as an observation into a histogram that represent places in the region. The problem then reduces to comparison of the histograms. We consider several measures based on normalized frequency vectors: L_1, L_2, L_∞, ChiSquared, Bhattacharyya and Kullback and Leibler divergence. Fuzzy histograms are also considered.

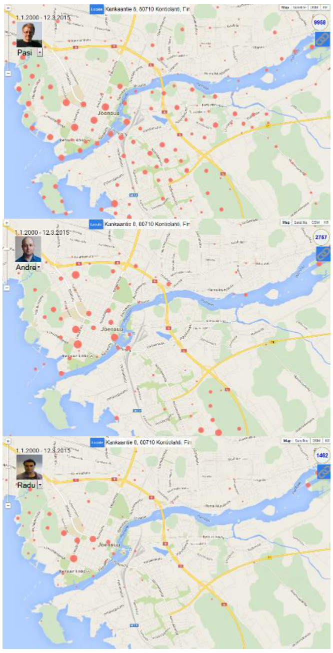


Fig. 1. Activity data of three users during 2011-2014.

2. Similarity of sparse location histograms

Mopsi (http://cs.uef.fi/mopsi/) is a prototype media sharing platform in which users can collect geo-tagged photos, track their routes, and recommend places of interest (Waga et al., 2012) to other users by upgrading them as *services*. These services include hotels, restaurants, cafeterias and many others that the users consider relevant to themselves and to others. The locations of these services serve as the histogram bins, which we denote as *places*. User similarities are calculated based on the locations where the users have collected data. We call these as *activity points*.

2.1 Locations, distance and places

Calculating similarity of two users starts by constructing histograms of the users. We process the activity points of a user by mapping them to their nearest *place*. Every activity point adds the count of the corresponding histogram bin by one. Distance calculation is based on *haversine* distance of the two locations given as latitude (φ) and longitude (λ) coordinates. The formula to calculate the haversine distance (in kilometers) is defined as:

$$hav = 2 \cdot R \cdot \arcsin) \tag{1}$$

$$\Psi = \sqrt{sin^2(\alpha) + \cos(\phi_1)\cos(\phi_2)\, sin^2(\beta)} \tag{2}$$

$$\beta = \frac{\phi_2 - \phi_1}{2}, \qquad \alpha = \frac{\lambda_2 - \lambda_1}{2} \tag{3}$$

where R=6372.8 km, φ_1 and φ_2 are the latitudes, and λ_1 and λ_2 are the longitudes of the two points. User has n activity points mapped into m histogram bins $h(i)$ so that:

$$\sum_{i=1}^{m} h(i) = n \tag{4}$$

The histograms are normalized as follows:

$$p(i) = \frac{h(i)}{\sum_{j=1}^{m} h(j)} \forall i \in [1, m] \tag{5}$$

where $p(i)$ represent the probability that an activity point belongs to the bin i. Example of the histogram construction is shown in Fig. 2, where three users have n_1=9, n_1=7 and n_1=7 activity points (small icons on map) are mapped to m=8 places (shown as the thumbnail images). The values of the bins (visit frequencies) are shown below each place.

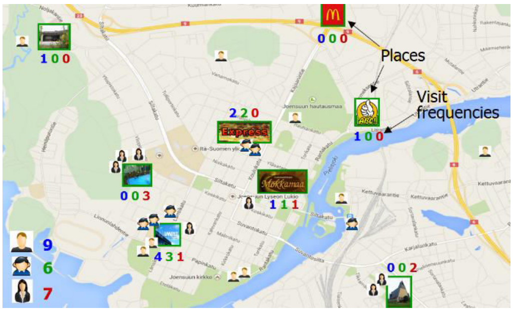


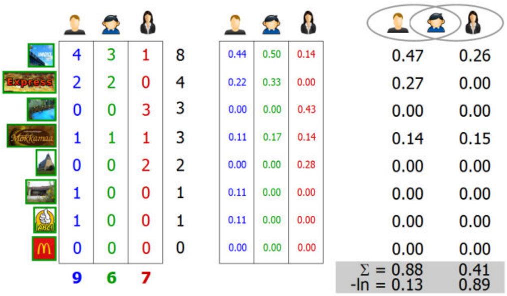
Fig. 2. Converting location history of three users to histogram consisting of m=8 predefined places.

2.2 Histogram comparison

The problem is how to calculate similarity (or distance) between two probability distribution functions (pdf), represented as histograms. The histogram is usually a one-dimensional array consisting of numerical values, for example, pixel intensities of an image. In this case, there is explicit ordering of the bins, and the values of the neighboring bins highly correlate with each other. Mapping an observation into the histogram is straightforward.

The bin values can also be nominal, or multivariate as in our case, so that there does not exists any natural ordering of the bins. However, since the observations appear in a metric space, the observations (activity points) can still be mapped to the histogram by simple distance calculations. The problem therefore reduces to histogram comparison, and there exists an extensive literature of different methods to solve the problem (Cha et al., 2007, Cha, 2008).

Fig. 3 demonstrates the process with our toy example, when using so-called *Bhattacharyya coefficient* originally proposed as a similarity measure between statistical populations. First, product $p_i{\cdot}q_i$ of two frequencies are calculated, and their square roots are then summed over the all histogram bins. The higher the frequencies, the higher the product. The result is converted to a distance in range [0,1] by applying logarithmic scaling. This provides natural bounds on the Bayes misclassification probability.



	4	3	1	8	0.44	0.50	0.14	0.47	0.26
Express	2	2	0	4	0.22	0.33	0.00	0.27	0.00
	0	0	3	3	0.00	0.00	0.43	0.00	0.00
Mokkamaa	1	1	1	3	0.11	0.17	0.14	0.14	0.15
	0	0	2	2	0.00	0.00	0.28	0.00	0.00
	1	0	0	1	0.11	0.00	0.00	0.00	0.00
	1	0	0	1	0.11	0.00	0.00	0.00	0.00
	0	0	0	0	0.00	0.00	0.00	0.00	0.00
	9	6	7					Σ = 0.88	0.41
								-ln = 0.13	0.89

Fig. 3. Example distance calculations of two histograms using Bhattacharyya distance.

Other commonly used distances considered here are various *L*-norms such as L_1, L_2 and L_∞, and classical Chi Squared distance. Kullback and Leibler distance (Ying et al., 2010) generalizes Shannon's concept of probabilistic uncertainty called entropy by calculating minimum cross entropy of two probability distributions (Liu and Schneider, 2012). These and the Bhattacharyya coefficient are defined below.

$$L_1 = \sum_i |p_i - q_i| \tag{6}$$

$$L_2 = \sum_i (p_i - q_i)^2 \tag{7}$$

$$L_\infty = max|p_i - q_i|\forall i \tag{8}$$

$$d_{ChiSq} = \sum_i \frac{(p_i - q_i)^2}{p_i + q_i} \tag{9}$$

$$d_{KLD} = \sum_i \left(p_i \cdot \log\frac{p_i}{q_i} + q_i \cdot \log\frac{q_i}{p_i} \right) \tag{10}$$

$$S_{BC} = \sum_i \sqrt{p_i \cdot q_i} \tag{11}$$

where p_i and q_i are the relative frequencies of the histogram bins *i*, and the summation is done over all the *m* places.

All the above techniques calculate the distance of each bin independently. In case of sparse observations, it may happen that strongly peaked histograms would become mismatched due to slight translation. So-called *earth mover distance* (EMD) (Rubner et al., 1992) aims at solving this by transforming surplus from one bin to the bins that have deficit. In case of one-dimensional numeric data this is straightforward to calculate by processing the histogram from sequentially from left to right. However, in multivariate case the optimal moving of the surplus becomes more complicated problem. It was noted in (Cha and Srihari, 2008) that the problem could be solved as transportation problem but faster algorithms would be needed.

In (Strelkov et al., 2008), the peaks of the histograms were considered as more important. Improved performance of L_1, L_2 and EMD was demonstrated in case of time-series analysis by calculating the sum of the peak weights multiplied by their closeness factors.

Sparseness of the data may also cause problems when there are too few observations compared to the number of available bins. Fuzzy histograms were proposed in (Fober et al., 2010) motivated by its successful application in image processing field. In case of one-dimensional histograms, observations are divided into several neighboring bins. We generalize the idea into multivariate case by utilizing the *k-nearest neighbors* (kNN) concept as follows.

For each location activity, we find its *k* nearest places and calculate fuzzy counts similarly as done in the well-known fuzzy C-means algorithm (Bezdek et al., 1984):

$$w_i = \left(\sum_{j=1, j\neq i}^{k} \frac{d(x - h_i)}{d(x - h_j)} \right)^{-1} \tag{11}$$

Otherwise, the histogram comparison can be done in the same way as with the crisp variant.

3. Experiments

We used data from Mopsi that has been collected using native mobile application available in all platforms with the following user distribution: WindowsPhone (55%), Android (28%), iOS (14%) and Symbian (3%). There are 36243 photos and 8963 trajectories, and by 9.5.2015, there have been 909 registered mobile users. Of these, 94 users have collected more than 5 photos and 5 routes. In the following, we focus on a small subset of users whose location activity we are familiar with.

3.1 Data extraction

For creating the histogram bins, we selected 293 services from Mopsi (http://cs.uef.fi/mopsi/). These include cafes, restaurants, holiday resorts, shops, parks and other services within the bounding box that covers the Joensuu sub-region, see Fig. 4. This region covers Joensuu downtown, its suburbans, neighboring municipalties and large sparsely inhabited rural areas. We use here only the location of the services. The coverage of the bins is dense in the downtown area but sparse in rural area. For example, Salomökki is the only service within 20 km radius whereas the core downtown there are about 75 services within the 3×3 blocks (300m×500m area) around the market place.

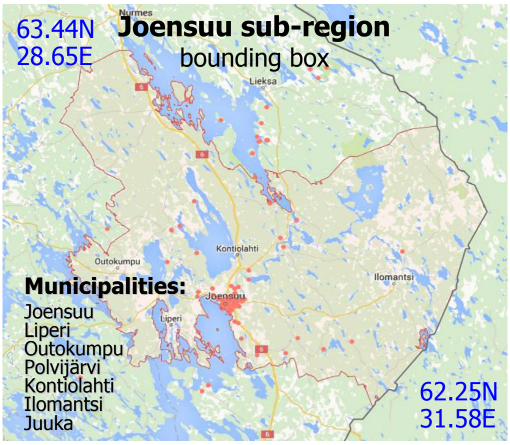


Fig. 4. Distribution of the places (histogram bins).

We then recorded activities of the users from the years 2011-2014 as follows: (1) places where they took photos, (2) places where tracking a route was started or ended. Each activity is counted to the frequency of its nearest service (histogram bin). Our first test consists of three active users (A=Andrei, P=Pasi, R=Radu) called A-P-R trio in the following. These users have 5831 location points in total. The most popular places with the corresponding visit frequencies are listed in Table 2.

The data is divided per year resulting into four subsets for each user, see Table 1. In total, we will have 12 subsets (pseudo users) denoted as: A11, A12, A13, A14, P11, P12, P13, P14, R11, R12, R13, R14. By default, the subsets A11-A14 should be similar to each other, and dissimilar to the other subsets. In practice, the situation is more complicated. The biggest frequency is in the bin corresponding to the location of user's home. However, both Andrei and Radu moved in 2014 causing significant changes in their histograms. All users have the same work place (Science Park). This and the homes are marked as underlines in Table 2.

Table 1. Three test users and the summary of their location data.

	2011	2012	2013	2014
Andrei	**206**	**757**	**432**	**329**
Pasi	**1263**	**545**	**636**	**751**
Radu	**37**	**292**	**324**	**259**

3.2 Test setup and results

For the histogram comparison, we calculate the similarity (or distance) value between all pairs of these 12 subsets. The task is then to decide which of the subsets belong to the same user by thresholding. The expected result is that 3·4·4=48 pairs (33%) should be recognized as the same user, and 3·4·8=96 pairs (67%) should fail the test.

For thresholding, we study the effect of different alternatives. First choice (average) is to use the average of all similarity values. This is the simplest non-parametric threshold that attempts to adapt the method to the data. Second choice (apriori) is obtained by selecting the threshold value (for the particular method) that passes 48 pairs (or as close to this as possible) based on a priori information that 33% values should pass the similarity test, and 96 should fail. The last choice (oracle) is the threshold that provides best accuracy for the particular method.

The results in Table 3 show that L_1, Chi^2, BHA and KLD provide good results (8%, 8%, 10%, 11%) using the average as threshold, and only slightly worse than if the optimal threshold (Oracle) was known (7%, 7%, 8%, 10%). The two other methods, L_1 and Chi^2, perform poorly (35%, 43%). The a priori information does not help, and even the result with optimal threshold is inferior (15%, 18%).

Fuzzy histograms were also considered with neighborhood of fixed size k=3. The classification errors systematically increased without clear reason. Especially KLD works significantly worse when using the fuzzy counts than with the crisp counts. Another observation is that the choice of the threshold becomes now critical. Using average as the threshold is no longer working with most measures.

The BHA and Chi^2 measures made classification error with users A13 and R13. In this case, the users reported to have recorded lots of joint bicycle trips at that year. In the data, this shows as increased counts in the bins representing rural areas.

Analysis of the other classification errors reveals the following details. All methods recognize A14 as different user than A11, A12 and A13. The same happens also between R11 and R14 with all methods except KLD. Both users changed their homes in 2014 causing different histogram bins to become dominant within the user. We therefore hypothesized that the methods might be affected too much by the dominant values. To test this, we performed additional tests by removing the top-10 location

Table 2. Ten most frequent entries in the histograms (first 10 entries).

	Andrei				Pasi				Radu				
	2011	2012	2013	2014	2011	2012	2013	2014	2011	2012	2013	2014	
Niinivaara otto3	20	0	29	150	47	7	6	8	1	2	2	3	275
Salomökki	13	11	87	36	64	17	16	7	0	1	12	2	266
keskusta 1	0	0	0	0	51	54	54	69	0	1	0	0	229
Skarppi – sauna	12	107	87	0	0	0	1	2	0	0	2	0	211
Noljakan kirkko	1	0	0	1	34	9	20	11	0	0	52	54	182
Lounasravintola Puisto	6	29	10	3	6	2	1	3	7	54	35	16	172
Joensuu Areena	7	92	6	0	18	4	7	15	0	13	4	2	168
Science Park	22	6	5	4	36	9	6	18	1	12	11	21	151
Kiesa	0	3	4	0	0	0	0	1	12	82	41	7	150
Oskolan lomamökit	0	0	0	0	73	6	48	13	0	0	0	0	140

Table 3. Classification accuracy for the APR trio.

	Threshold (Crisp)			Accuracy (Crisp)			Accuracy (fuzzy)		
	Average	Apriori	Oracle	Average	Apriori	Oracle	Average	Apriori	Oracle
L1	0.31	0.27	0.28	8%	8%	7%	10%	10%	10%
Chi2	1.22	1.24	1.18	8%	7%	7%	17%	11%	10%
BHA	0.46	0.46	0.48	10%	10%	8%	15%	14%	11%
KLD	0.82	0.89	0.88	11%	11%	10%	36%	21%	15%
L2	0.84	0.80	0.88	35%	47%	15%	35%	49%	14%
L∞	0.79	0.72	0.87	43%	43%	18%	38%	47%	21%

values (those shown in Table 3). The changes are summarized in Table 4. The revised experiments show even weaker performance. The changes did not make A14 to match with A11-A13. On the contrary, it caused more mismatch results and showed, that the methods somewhat rely on the detection of user's most popular working place, which is not very encouraging for the capability to distinct

Table 4. Classification accuracy for the APR trio when 10 most popular bins have been eliminated from the calculations.

Method:	Change:	Observation:
L_1	8% → 24%	Loses its ability
Chi^2	8% → 13%	A11 becomes similar with P11, P13, P14
BHA	10% → 13%	A11 becomes similar with P11, P13, P14. P11-R14 no longer match.
KLD	11% → 13%	A11 and R14 become similar, no other effects.
L_2	35% → 40%	Works even worse.
L_∞	42% → 43%	Works even worse.

3. Conclusions

Locations of people show their preferences and interests. In this paper, we present how to find similar users based on their location history. We have shown that L_1, ChiSquared, Bhattacharyya and Kullback and Leibler divergence are all applicable as histogram comparison methods for the problem. Fuzzy histograms, however, worked worse than crisp histograms.

As future work, we consider the following open questions. Automatic method for selecting the threshold should be constructed based on expected distribution, as well as optimizing the number of bins. Study whether double normalization, log-scaling of frequencies, using cosine distance, and fuzzy modeling for sparse data can provide improvement. Study the size of test data vs. recognition accuracy. Generalization of earth mover distance for the multivariate case is worth consideration. Finally, histogram-based comparison might be better to replace prototype based comparison using clustering. These are all points of future studies.

References

Adomavicius, G., Tuzhilin, A., 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, *IEEE Transactions on Knowledge and Data Egineering*, 17 (6), 734-749.

Bao, J., Zheng, YH., M.F. Mokbel, 2013. Location-based and preference-aware recommendation using sparse geo-social networking data, *Int. Conf. on Advances in Geographic Information Systems* (SIGSPATIAL), 199-208, Redondo Beach, CA.

Bezdek, J. C., Ehrlich, R., Full, W., 1984. FCM: The fuzzy *c*-means clustering algorithm, *Computers & Geosciences*, 10 (2-3), 191-203.

Biagioni, J., Krumm, J., 2013. Days of our lives: Assessing day similarity from location traces, *User Modeling, Adaptation, and Personalization*, LCNS vol. 7899, pp. 89-101. Springer Berlin Heidelberg.

Cha, S.-H., 2007. Comprehensive survey on distance/similarity measures between probability density functions, *Int. J. Mathematical Models and Methods in Applied Sciences*, 4 (1), 300-307.

Cha, S.-H., Srihari, S. N., 2008. On measuring the distance between histograms, *Pattern Recognition*, 29 (13), 1768-1774.

Cha, S.-H., 2008. Taxonomy of Nominal Type Histogram Distance Measures, *American Conf. on Applied Mathematics*, 325-330, Harvard, MA, USA.

Chen, X., Pang, J., Xue, R., 2013. Constructing and comparing user mobility profiles for location-based services, *ACM Symposium on Applied Computing*, pp. 261-266.

De Pessemier, T., Minnaert, J., Vanhecke, K., Dooms, S., Martens, L., 2013. Social Recommendations for Events, ACM Conf. on Recommender Systems, Hong Kong, China.

Fober, T., Hullermeier, E., 2010. Similarity measures for protein structures based on fuzzy histogram comparison, *IEEE Int. Conf. on Fuzzy Systems*, 1-7, Barcelona.

Guy, I, Jacovi, M., Perer, A., Ronen, I., Uziel, E., 2010. Same places, same things, same people?: mining user similarity on social media, ACM conference on Computer supported cooperative work, 41-50, Savannah, GA, USA.

Li, X., Guo, L., Zhao, Y., 2008a. Tag-based social interest discovery, *Conf. on World Wide Web*, 675-684, Beijing, China.

Li, Q., Zheng, Y., Xie, X., Chen, Y., Liu, W., Ma, W.-Y., 2008b. Mining user similarity based on location history, *ACM SIGSPATIAL Int. Conf. on Advances in geographic information systems*, Paper No. 34, Irvine, CA, USA.

Liu, H., Schneider, M., 2012. Similarity measurement of moving object trajectories, *Int. Workshop on GeoStreaming*, 19-22.

Fränti, P., Waga, K., Khurana, C., 2015. Can social network be used for location-aware recommendation?, *Int. Conf. on Web Information Systems & Technologies* (WEBIST'15), Lisbon, Portugal.

Rubner, Y., Tomasi, C., Guibas, L. J., 1992. A metric for distributions with applications to image databases, *IEEE Int. Conf. Computer Vision*, 59-66.

Strelkov, V. V., 2008. A new similarity measure for histogram comparison and its application in time series analysis, *Pattern Recognition Letters*, 29 (13), 1768-1774.

Waga, K., Tabarcea, A., Fränti, P., 2012. Recommendation of points of interest from user generated data collection, *IEEE Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing* (CollaborateCom*),* Pittsburgh, USA.

Wang, H., Liu, K., 2012. User oriented trajectory similarity search, *Int. Workshop on Urban Computing*, 103-110.

Yang, X., Steck, H., Guo, Y., Liu, Y., 2012. On Top-k Recommendation using Social Networks, ACM Conf. on Recommender Systems, Dublin, Ireland, 67-74.