Michael Oliveira da Cruz

On the similarity of users in carpooling recommendation computational systems

São Cristóvão

February 26th, 2016

Michael Oliveira da Cruz

On the similarity of users in carpooling recommendation computational systems

Thesis submitted to the Graduate Program in Computer Science of the Federal University of Sergipe in partial fulfillment of the requirements for the Degree of Master of Science in Computer Science.

Federal University of Sergipe – UFS

Department of Computer Science

Gradute Program in Computer Science - PROCC

Supervisor: Hendrik Teixeira Macedo

São Cristóvão

February 26th, 2016

Michael Oliveira da Cruz
On the similarity of users in carpooling recommendation computational systems/ Michael Oliveira da Cruz. – São Cristóvão, February 26th, 2016-
83 p. : il. (algumas color.) ; 30 cm.

Supervisor: Hendrik Teixeira Macedo

Dissertação (Mestrado) – Federal University of Sergipe – UFS
Department of Computer Science
Gradute Program in Computer Science - PROCC, February 26th, 2016.

1. Palavra-chave1. 2. Palavra-chave2. 2. Palavra-chave3. I. Orientador. II. Universidade xxx. III. Faculdade de xxx. IV. Título

Michael Oliveira da Cruz

On the similarity of users in carpooling recommendation computational systems

Thesis submitted to the Graduate Program in Computer Science of the Federal University of Sergipe in partial fulfillment of the requirements for the Degree of Master of Science in Computer Science.

Approved work. São Cristóvão, February 26th, 2016:

Dr.Hendrik Teixeira Macedo
Advisor

Dr. Alberto Costa Neto
Comitee member (Internal)

Dr. Luciano de Andrade Barbosa
Comitee member (External)

São Cristóvão
February 26th, 2016

To my mother, my first example of persistence and dedication.

Acknowledgements

First I thank God who has guided my life and has given wisdom, patience and strength.

I would also like to thank my family, especially my sisters and my brother Edivan Oliveira who always have supported and guided me.

The Department of Computer Science at the Universidade Federal de Sergipe-UFS, for supporting me.

I would also to thank my advisor, Prof. Dr. Hendrik Teixeira Macedo who guided me through this wonderful research. Every suggestion, review, comment were important to my formation.

I also like to thank Prof.Msc.Adolfo Pinto Guimarães for their support, patience and suggestions.

I also thank the teachers who somehow contributed to my academic education and the master classmates by fellowship and the hard times that have brought us knowledge.

I also like to thank Erick Mendonca and Raphael Barreto for the help in the development of applications of this work.

Finally, the authors thank Fundação de Apoio à Pesquisa e Inovação Tecnológica do Estado de Sergipe (FAPITEC-SE)for granting a scholarship and the Universidade Federal de Sergipe for the financial support [Edital POSGRAP/COPES/UFS No 03/2014 14/2012 (HERMES),Processo 008325/14-72].

"Thus, the task is not so much to see what no one yet has seen, but to think what nobody yet has thought about that which everybody sees." (Arthur Schopenhauer)

Abstract

Problems related to urban mobility is a big concern to public administration. Some policies have been adopted in order to soften those problems in large cities. Building new routes, encouraging the use of public transportation, building new bike paths and encouraging the use of bicycle are some of them. A common practice which is closely related to cultural habits in some nations and which can contribute to soften the problem is ridesharing. Ridesharing is defined as a grouping of travellers into common trip by car or van. Though there exist some applications that aim to facilitate the practice of ridesharing, none of them have the functionality to search automatically for users with similar trajectories or demographic and social profile. In this work, we proposed an innovative approach, considering ridesharing context, that aims to discover clusters of users that have similar trajectories, clusters of users that have similar profile and clusters of users with similar trajectory and similar profile. Furthermore, we define a formalization of ridesharing terms and an initial time complexity analysis is done. A social network for ridesharing has been also modeled and developed according to proposed approach. Experimentation and evaluation method consist of: (i) Building a dataset from volunteers in transit with GPS-equipped smartphones, (ii) Using proposed approach to generate clusters of users and application of Davies-Boulding index metrics which reflects how similar the elements of the same cluster are, as well as the dissimilarity among distinct clusters. Results show the feasibility of the approach to problem solution if compared with some approach established in literature such as, K-means. Results of dataset analysis show that some traffic information should undergo data mining. Finally, social network mobile app acceptance was measured by questionnaire.

Keywords: Urban Mobility, ridematching, ridesharing, carpooling, clusters, Gps.

Resumo

A falta de mobilidade urbana é uma grande preocupação da gestão pública em todo o mundo. Algumas políticas têm sido adotadas a fim de minimizar seus efeitos nas grandes cidades. Construção de rotas alternativas, melhorias e incentivo ao uso de transportes públicos, construção de ciclovias e estímulo ao uso de bicicletas são algumas dessas políticas. Uma prática que pode contribuir para a solução do problema é a carona. Carona consiste no ato de transportar gratuitamente num veículo pessoas que possuem trajetórias semelhantes. Embora existam algumas aplicações que se propõem a facilitar a prática de caronas, nenhuma dessas aplicações possuem funcionalidades de busca por usuários que possuem similaridades de trajetória e de perfil demográfico e social. Neste trabalho, propomos uma abordagem inovadora, considerando peculiaridades do contexto do uso de caronas, que visa a descoberta de agrupamentos de usuários que possuem trajetórias semelhantes, usuários que possuem perfis semelhantes e agrupamentos de usuários que são similares considerando suas trajetórias e seus perfis. Elementos intrínsecos ao problema são formalmente definidos e uma primeira análise de complexidade para tempo de processamento foi realizada. Uma rede social de propósito específico para o compartilhamento de caronas foi modelada e implementada com respeito à abordagem proposta. O método para experimentação e avaliação da abordagem consistiu (i) na confecção de base de dados alimentada periodicamente em tempo real por dados de trânsito obtidos a partir de aparelhos de *smartphone* com GPS de voluntários em trânsito com seus automóveis, (ii) aplicação da abordagem proposta para geração dos agrupamentos de usuários a partir da base estabelecida e (iii) aplicação da métrica *Davies-Boulding Index*, que indica o quão factível os agrupamentos são. Resultados mostraram a efetividade da abordagem para solução do problema se comparada a formas bem estabelecidas da literatura relacionada, como o K-means, por exemplo. Resultados da análise da base de dados também mostraram que algumas informações de trânsito podem ser inferidas a partir de ações de mineração. Por fim, a aceitabilidade de potenciais usuários da rede social foi medida a partir de questionário.

Palavras-chave: Mobilidade Urbana, ridematching, caronas, agrupamento, Gps.

List of Figures

List of Tables

List of abbreviations and acronyms

WHO	World Health Organization
LCSS	Longest common subsequence
EDR	Edit distance
ICT	Information and Communication Technologies
GPS	Global Position System
POIs	Point of Interests
UMD	University of Maryland
MVC	Model-View-Controller
REST	Representational State Transfer
GLN	Natural Language Generation
W3C	World Wide Web Consortium
WCAG	Web Content Acessibility Guidelines
DBI	Davies-Boundin index
NC	Number of Clusters
MinPts	Minimum Number of Neighbours
HOV	High-occupancy Vehicle
LD	Long Distance
SD	Short Distance
CG	Carpool Group
CSPA	Cluster-based Similarity Partitioning Algorithm
HPGA	HyperGraph Partitioning Algorithm
MCLA	Meta-Clustering Algorithm
NFC	Number of Final Clusters

DBIT	Davies-Boulding Index Related to Trajectory
DBIP	Davies-Boulding Index Realted to Profile
CNG	CG that cannot provide a vehicle

List of symbols

$\leq$	Less than or equal to
ε	Epsilon
δ	Delta
ϕ	Phi
$\in$	Element of
$\geq$	Greater than or equal to
$\sum$	Sigma
α	Alpha
$\sim$	Similarity
$\simeq$	Approximately equal
θ	Theta
Tr	Trajectory
$R()$	Ride
a	Passenger
d	Driver
$V()$	Vehicle
A	Set of Passenger
τ	Tau
$\cap$	Intersection

Contents

1 Introduction

Urban mobility can be described as a key to dynamic urbanization. According to Stafford dictionary, mobility is defined with the ability to easily move or travel around. Urban mobility has been a subject of a lot of research (HE; HWANG; LI, 2014a),(AGATZ et al., 2012),(FUNK, 2015), because there exists problems which are causing a lot of losses such as, traffic congestion, environmental damages, health issues etc. Traffic congestion is one of the main problem that decreases the quality of urban mobility and it is a reality in most cities around the world. (LERNER; AUDENHOVE, 2012) make a prevision that, in 2050, 6.3 billion of people will be living in urban areas and consequently the amount of trips will be the triple.

Many factors can contribute to congestion in large cities, but the increasing number of vehicles can be identified as the main factor. According to the National Traffic Department, the number of vehicles in Brazil increased by over 110% in recent years (DENATRAN, 2013). The growth in the number of private cars has caused a lot of problems, indeed. Most Brazilian cities do not have a proper road structure to accommodate the number of vehicles. These factors contribute to the increasing number of congestion in these cities.

The traffic congestion has caused expressive economic losses and reflects on various aspect of society. Home delivery of foods, for instance, is directed affected by congestion and passes on the costs to the population. According to (SCHRANK; EISELE; LOMAX, 2015) which has produced the US urban report information about congestion levels in 2014, the costs as a result of traffic congestion has increased. In 2000, for instance, $114 billion were spent and 2014, $160 billion. The congestion wastes were 6.9 billion hours extra time, peoples have wasted 42 hours traveling compared to 18 hours in 1982. European Commission[1] estimates that the road congestion costs nearly €100 billion every year. According to (CINTRA, 2014) the costs with congestion in São Paulo is nearly of R$ 40 billion by year.

Apart from economic losses, environment damages are a serious problem. The amount of pollution or greenhouse gases generated by cars contributes to climate changes and to health issues like asthma, allergic rhinitis and atopic dermatitis. According to (LEVY; BUONOCORE; STACKELBERG, 2010), the impact of fine particulate matter (PM10) in mortality in 2005 was 3000 premature deaths. (YANAGI; ASSUNÇÃO; BARROZO, 2012) shows that PM10 can contribute to increased incidence of some cancers.

The total amount of fatal accidents that occurs in traffic is more one concern.

[1] http://ec.europa.eu/transport/themes/urban/doc/ump/flash-eurobarometer-ump-2013.pdf

According to WHO (World Health Organization)[2], 1.24 million road traffic deaths occurred in 2013. WHO presents that around 186.300 children under 18 years die from road traffic crashes annually and the rate of road traffic death are three times higher in developing countries. In Brazil, 42.226 deaths occurred in 2013 according to Ministério da Saúde[3]. WHO foresee that the situation related with road traffic death will worst in the next years, because of the increase number of car in road traffic and the lack of city planning etc.

Thus, strategies to reduce the amount of vehicles in cities should be thought. In Beijing, China, where traffic is considered to be some of the world's worst, government has adopted the policy of restricting traffic for private cars. Even with this policy, though, traffic condition in peak hours is critical (HE; HWANG; LI, 2014b). The city of São Paulo has adopted similar policy (CET, 2013), but even so, it had some of the worst episodic of vehicle congestion in 2013 (G1, 2013). The use of alternative means of transport such as bicycles and subways is an option.

The Brazilian government created a bill called National police of urban mobility[4]. The government intends to stimulate the use of public transportation in order to decrease the amount of car in roads traffic. According to the bill, the cities will have to provide an urban mobility planning project within three years and such planning must improve the transportation system and the infrastructure of the roads. Furthermore, the planning will foresee some initiatives such as ways to facilitate the access to the public transportation, the build of new roads and the encouragement of bicycle usage etc.

Another alternative to improve the quality of urban mobility is to encourage the use of carpooling or ridesharing, which consists of sharing private vehicle space among people with similar destinations or daily trajectories (GOWRI, 2008). Sharing cars' empty seats is indeed such a kind of optimization procedure if we consider, for instance, the low occupancy rate per vehicles in traffic (HE; HWANG; LI, 2014a). The mean occupancy of people per vehicle in U.S.A. transit in 2001 was 1.6. More recently, in 2011, a research conducted by Michigan University has shown a occupancy rate of 1.5. Such occupancy rate is easily decreased to 1.4 when the trajectory is limited to "house-work" or "work-house". In other words, there are plenty of vehicles with just the driver inside (GHOSEIRI et al., 2011). It is possible to conclude that the use of empty seats in vehicles might be an effective way to increase occupancy rate and as a result to soften traffic congestion. However, the practice of ridesharing is closely related to cultural habits. In Europe, for example, the practice of get a ride is more common than in Brazil. But, even in some countries as England, according to Statista [5] the average of rate occupancy between car

[2] http://www.who.int/roadsafety/week/2015/en/
[3] http://www.vias-seguras.com/os_acidentes/estatisticas/estatisticas_nacionais/estatisticas_do_ministerio_da_saude
[4] http://www.senado.gov.br/noticias/Jornal/emdiscussao/motos/legislacao-e-fiscalizacao/politica-nacional-de-mobilidade-urbana-pnmu-do-governo-federal-lei-12-587-12-pretende-estimular-transporte-coletivo-publico-nas-cidades.aspx
[5] http://www.statista.com/statistics/314719/average-car-and-van-occupancy-in-england/

and van is just 1.5 (considering the period of 2002 to 2014).

The practice of ridesharing in Brazil isn't so common yet, but there is a growing concern with the creation of new tools that promote the efficient usage of transport in order to avoid traffic jam and, as a consequence, provide a better environmental quality. There exists some software initiatives to facilitate the ridesharing's practice such as Caronas Brasil (AZZAM; BELLIS, 2008), Blablacar (MAZZELLA, 2004), Tripda (VAXMAN et al., 2014), Uber (KALANICK; CAMP, 2015), Zaznu (FABER, 2014), Lyft (ZIMMER, 2015), RideWith (BARDIN et al., 2015). But, even with the release of some application and the proliferation of smartphones, GPS and others technologies, the New York Times reported, in 2011, that ridesharing has continued to decline in US (FURUHATA et al., 2013).

Although there is a growing number of related applications been released, it is important to note that some applications like Uber, Zanznu work as much like a taxi, and the idea of ridesharing does not seem to actually apply in such cases. Moreover, some services provided by these systems require that interested users perform a search for people who offers a ride with the same or similar trajectories. In addition to the inherent difficulty in finding a corresponding ride, the driver and passengers are often unfamiliar which leads to safety concerns. Finally, there is also no easy way to choose or filter trip coleagues according to their social or demographic profile.

(FURUHATA et al., 2013) considers integrating the ridematching system with social networking sites that enable users to obtain more background information of potential drivers and passengers. To date, few ridesharing software have had commercial success (GHOSEIRI et al., 2011), because the applications usually don't provide important features such as safety, flexibility, efficiency and usability. Furthermore, other important factors may discourage the practice of ridesharing: smoking, features of the vehicle itself, social and demographic profile of the driver (AGATZ et al., 2012) and gender (LEVIN et al., 1977). The so-called ridematching procedure has been proposed to deal with these issues and suggest the ridesharing formation instantaneously (AGATZ et al., 2012). It promises easing the matching process among candidates by properly assigning users who wish to get a ride to users that offer.

Most research has focused on improvement of the trajectory mining process (HE; HWANG; LI, 2014c),(LEE; HAN, 2007), but, until now, no one has proposed an effective approach to integrate ridematching according to users trajectories and profile. Such an approach may increase the flexibility, efficiency and safety of ridesharing applications. Another research gap is the lack of formal definition related to elements of ridesharing context. Indeed, there are a plenty of different terms with same meaning such as route (HE; HWANG; LI, 2014b) or trajectory (LEE; HAN, 2007), passenger or riders (AGATZ et al., 2012) and so on.

Considering the strategy proposed by (FURUHATA et al., 2013), which integrates

ridematching applications with social network a way to encourage people to use ridesharing application, some questions arise, for example, which type of ridematching approach is necessary to develop similarity between trajectory of users? Another challenge is related with the type of data that can be used from social network in order to improve ridesharing application more safety? Other question to be considered is how to associate ridematching information with social information? The approach used in our work considers integrate ridematching information with user's profile information. So, developing ridematching approach is necessary to consider the similarity between trajectories. There exists some methods in literature that calculate similarity such as LCSS (Longest common sub sequence), EDR (Edit distance), but, in general, these approaches consider all points of the trajectory which is very time consuming. Other concern is related with the number of points that represent the trajectory; usually, when trajectories is represented a set of triple (latitude, longitude, time), if the time interval between successive collected points is to small, some way to eliminate redundant information while preserving trajectory integrity is highly demanding.

The similarity of profile in context of ridesharing hasn't been properly explored. The first concern is related to attributes that can be considered as relevant in ridesharing context. Another no less important concern is how integrate similarity of profile and trajectory together in order to discover a set of people who are similar. Discovering persons that share similar trajectory and profiles may contribute to reduce the hassle of people with regards to ridesharing apps.

Although the similarity of profile and trajectory is important in the context of ridematching, it's necessary consider that relevant information is achieved when there exist good datasets. Currently, there exists a plenty of research related to urban mobility (HE; HWANG; LI, 2014b), trajectory (LEE; HAN, 2007), carpooling (KELLEY, 2007), but there aren't enough available datasets. Such datasets may help researchers to understand the behavior of the cities. There exist some types of urban mobility datasets where the data is collected from a large amount of auxiliary instruments, such as cameras, inductive-loop detectors, GPS (Global Position System), microwave detectors, but these data are still scarce. Currently technologies like GPS and Wifi have enabled people to record their location history as a sequence of time-stamped locations (ZHENG; XIE; MA, 2010). GPS data has been used in some application which aim to perform trajectory mining or find similar trajectory, mining point of interests (POIs), find out sub-trajectories and so on. These applications can improve information about urban mobility of the cities and GPS devices are one of the cheapest way to collect data.

GPS and Internet-enabled computable devices are handy. Despite of the use of GPS and Internet as a effective tool to record people location history as a sequence of

time-stamped locations and some government initiatives to this end [6], these data are still scarce. Current research endorse such shortage as a severe limitation (HE; HWANG; LI, 2014a), (LEE; HAN, 2007), (CRUZ; MACEDO; GUIMARãES, 2015).

The goal of this dissertation is twofold: (i) provide and evaluate an innovative approach to perform ridemacthing considering trajectory similarity and user profile similarity, (ii) embed such approach into an actual application for ridesharing.

This dissertation is organized in self-contained chapters which present isolated results and contributions. In order to achieve the goals proposed, efforts have been made. Such efforts have generated submissions and publications. Chapter 2 presents an extension of the social network GO! (MATOS et al., 2014). The new social network called GO!Caronas remodeled GO! in order to support the approach presented in chapter 3. The motivation of this work is established as a strategy to join ridesharing system with a social network as a way to encourage people use carpooling system, because, though ridesharing applications have already existed, the number of users has decreased. The chapter shows the new technologies used to remodel the application and presents results of system's profiling. Such chapter also describes the architecture of the system and implementation details.

In chapter 3, we present an extension of the paper (CRUZ; MACEDO; GUIMARãES, 2015) that proposes an innovative ridematching approach that utilizes a method for trajectory discretization. Such paper presents a ridematching approach based on clusters. The clusters are generated based on Optics algorithm, which is a density-based algorithm. The method is composed of some steps, such as trajectory discretization, temporal filter and clustering. The extension focuses on the ridematching approach in order to consider user's profile and, consequently, increase the robustness of the method to find users that have similar trajectory and profile. The motivation of this approach is related to the possibility to give more information about who is getting or giving a ride. Besides the clusters algorithm, this work uses ensemble learning approach to merge trajectory clusters with profile clusters and the work also propose to formalize fundamental elements which make part of carpooling context. Some experiments are described in order to prove the feasibility of the approach. The results have showed that the proposed approach is feasible. This paper is the second step to achieve the first goals.

Chapter 4 proposes an open urban mobility dataset. This work presents an urban mobility dataset which was build and properly evaluated. Such paper also describes the GO!Track application which was developed to smartphones devices and shows the dataset structure, such as attributes, relationship and so on. Besides, the chapter presents some analysis of data and some applications of machine learning methods in order to verify if there exists possibility of discovering information about traffic. The dataset is currently receiving data and it was used to experiments performed by the approaches described in

[6] http://data.rio/dataset/GPS-de-onibus

chapter 3.

Finally, the pertinent conclusions and considerations are discussed in the chapter 5. The Figure 1 shows map mind of this work. The green arrows present the respective chapters and the gray arrows summarize the main subjects addressed in introduction. The blue arrows indicate there exist some relationship between chapters.

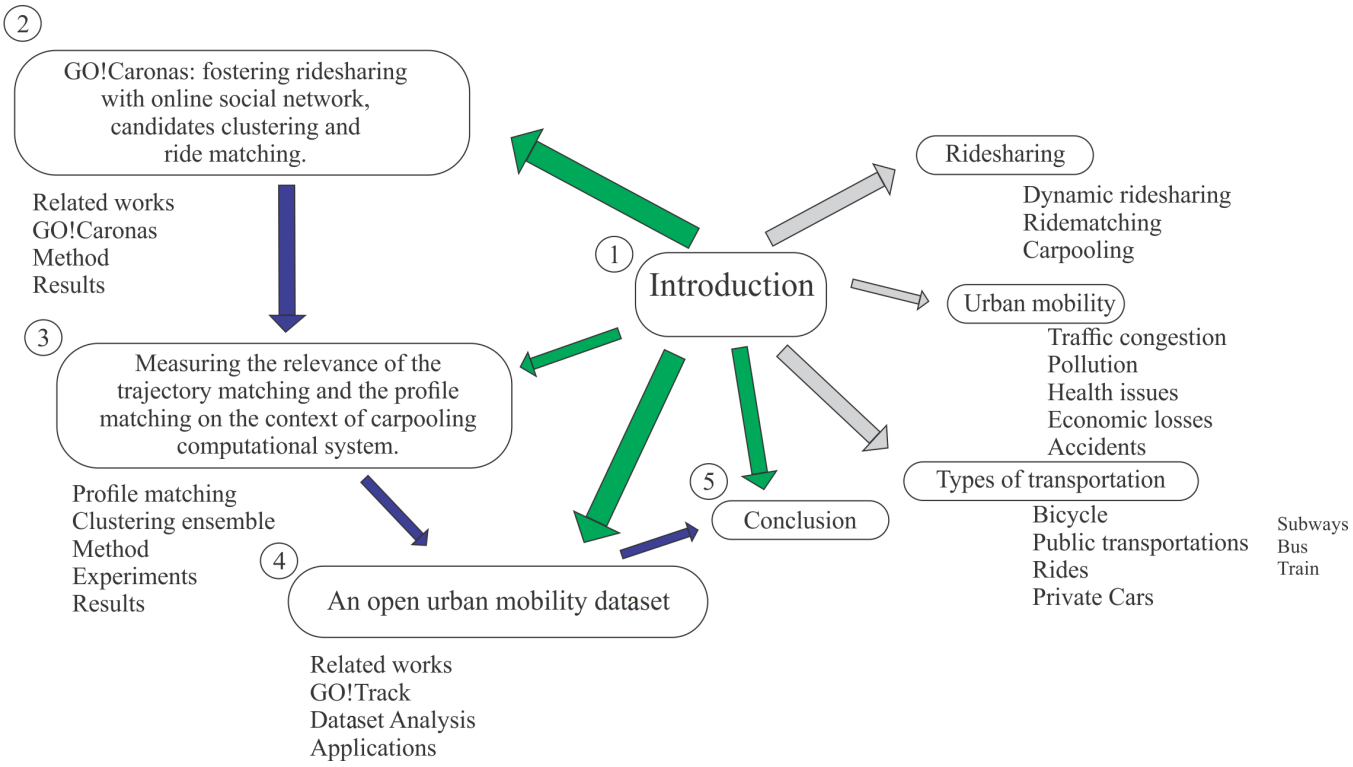


Figure 1: Figure shows the map mind of the work.

2 GO!Caronas: fostering ridesharing with on-line social network, candidates clustering and ride matching

This chapter proposes extending an on-line social network GO!(MATOS et al., 2014) to support the use of ridematching proposed by (CRUZ; MACEDO; GUIMARãES, 2015). This extension has been called GO!Caronas. The ridematching method is an innovative approach which permits to generate trajectory clusters based on POIs around the trajectory. Furthermore, GO!Caronas is integrated with a new functionality which permits users to form groups of users with similar trajectories.

The rest of this chapter is organized as follows. The Section 4.1 reviews systems with similar features. Section 4.2 presents the architecture and the main functionalities of the social network GO!Caronas. In section 4.3, we describe the ridematching approach. The Section 4.4 shows the results of the prospecting demand study and profile analysis of ridematching approach. Finally, in Section 4.5 we present some concluding remarks and future directions.

2.1 Related Works

Carpooling is a specialization of ridesharing. Ridesharing is defined as grouping of travellers into common trip by car or van (CHAN; SHAHEEN, 2012). There exist some specialization of ridesharing, for instance, casual carpooling, real-time ridesharing, social networks and so on. Casual carpooling normally is formed during morning commute hours at park-and-ride facilities or public transit centers and takes advantage of existing HOV(High-Occupancy Vehicle) lanes to get to a common employment centre. Real-time ridesharing is more flexible than casual carpooling, because it uses mobile applications and automated ridematching software to organize ride in real time. This enables participants to be organized either minutes before the trip takes place or while the trip is occurring. The last kind of ridesharing uses social networks to find or match potential riders among friends. Social network platform are used with expectation that users can build trust and safety estimation (CHAN; SHAHEEN, 2012).

Applications like Uber (KALANICK; CAMP, 2015), lyft(ZIMMER, 2015), Carticipate(FROST, 2015), Tripda (VAXMAN et al., 2014), BlaBlacar(MAZZELLA, 2004) and Carma (O'SULLIVAN, 2015) are considered real-time ridesharing, because they do not require that participants know each other, in other words, require little relationship

between participants and have some kind of ridematching method to find similar rides. Another characteristic is that most of these applications work like taxi and the idea the common trip is not so valid. Furthermore, some applications work like a kind of sophisticate taxi what is very different of ridesharing definitions.

The system CaronasBrasil (AZZAM; BELLIS, 2008) is one of the first systems for carpooling in Brazil. This application distinguishes of capooling definition gave by (CHAN; SHAHEEN, 2012), because it is more flexible, in other words, users can get or offer ride in any time of the day. The Figure 2 shows a kind of classification among applications cited.

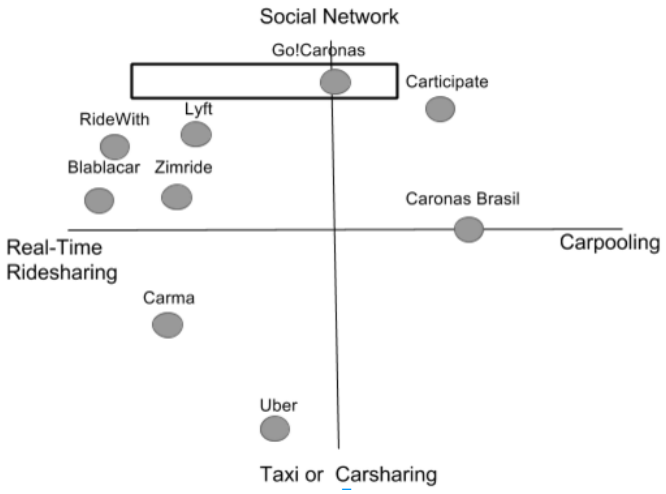


Figure 2: The comparison among applications.

The Figure 2 shows that great part of applications are classified as a real-time ridesharing and they have some of characteristics of social network, since those applications don't build, but use social networks like Facebook[1], Goolge+ [2] etc. GO!Caronas application tries to keep three characteristics that have been judged important according to (CHAN; SHAHEEN, 2012).

Another characteristics such as long distance (LD), short distance (SD), ridematching and profile matching can be other alternative to classify ridesharing applications. Long and short distance are related with length of trajectory that users normally share in some applications. Since ridematching and profile matching are characteristics related to the possibility that applications make matching automatically considering user's trajectory, user's profile or both. The Table 1 shows the classification of systems according to characteristics described.

GO!Caronas proposes to add ridematching approach to support real-time ridesharing, but the idea the common trip is maintained. Another functionality is to permit that

[1] www.facebook.com
[2] www.plus.google.com

Table 1: Ridesharing applications

Applications	LD	SD	RMatching	PMatching
Carma	Yes	Yes	Yes	No
Tripda	Yes	No	Yes	No
Uber	Yes	Yes	Yes	No
BlaBlaCar	Yes	No	Yes	No
Lyft	Yes	Yes	No	No
CaronasBrasil	Yes	No	No	No
Carticipate	Yes	Yes	No	No
GoCaronas!	Yes	Yes	Yes	No
RideWith	Yes	Yes	Yes	No

people may create group with regular schedule in order to preserve the characteristic of carpooling.

2.2 GO!Caronas

The system GO!Caronas extends the system GO! (MATOS et al., 2014) which is a social network that allows the sharing of rides among users. The GO!Caronas adds ridematching algorithm in order to turn the social network in real-time ridesharing. Now, it is possible organize ridesharing in real time, just minutes before the trip takes place. Besides ridematching, the functionality of group formation was added to permit users build groups. Groups is a interesting functionality because permit that users invite users that work or study in the same place etc.

2.2.1 Architecture

The architectural pattern used in the project is the MVC (Model-View-Controller). The MVC divide the project into three layers: (1) model, (2) view and (3) controller [9].

The model layer is responsible for providing all communication with the database (logic and business rule). The view layer presents information to the user. This layer is not concerned with any kind of data treatment, but only to present information and receive user input. The control layer performs some treatments on the data received by the view layer and specifies the sequence of views. It is also responsible for handling the communication between the model and view layers.

The Figure 2 shows the main components of the architecture. We can observe the three MVC components responsible for the requests related to the systems model, vision and control.

The system implements two types of controllers: (1) Web Controller, which is

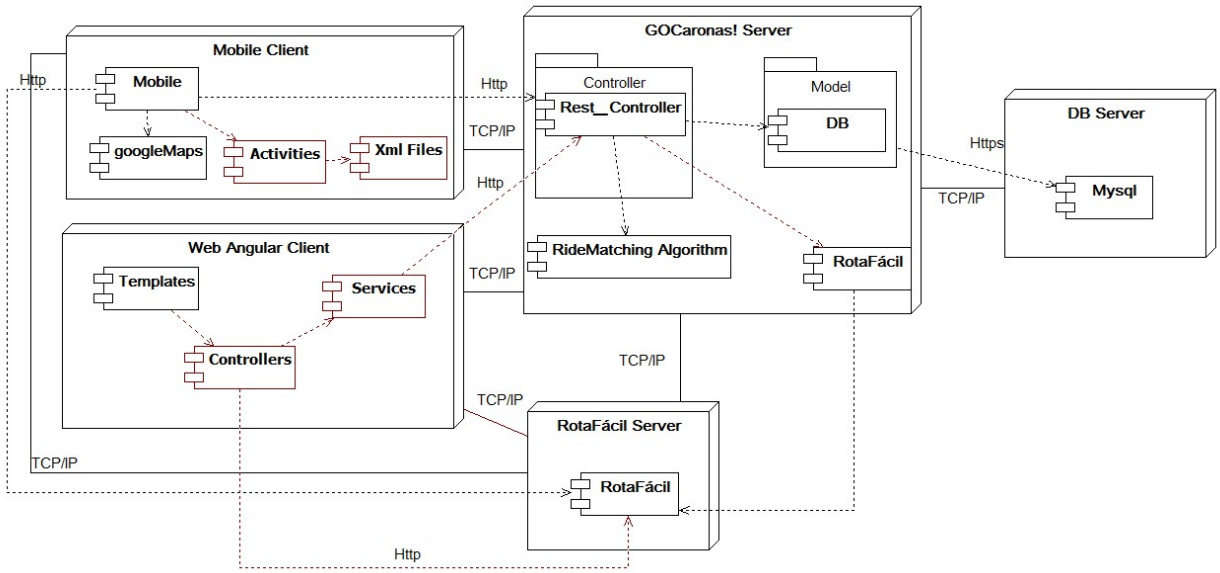


Figure 3: Architecture details GO!Caronas.

implemented with the angularjs [3], (2) Mobile Controller, both controllers send or receive data in JSON format. These formats allow for greater data security. This controller uses the REST (Representational State Transfer) architecture (ALLAMARAJU, 2010).

2.2.2 System Description

Table 2 lists the old and new functionalities identified for a ridesharing system. The old functionality is related to GO!(MATOS et al., 2014). These features have been identified through of analysis done in the major related systems. The Table 2 shows those that are available in the Web client and mobile client. An explanation of each functionality is presented below.

2.2.2.1 Ridematching

The driver must registrate a ride or a passenger must registrate some information, for example, departure and destination point, time and date. The ridematching algorithm will be capable of finding user with common interests, in other words, the system will find drivers and passengers with similar trajectory and schedule.

2.2.2.2 Groups

The users can create groups of people through ridematching algorithm or they can create manually. To create group through the algorithm, the users must registrate a

[3] https://angularjs.org/

Table 2: GO!Caronas functionalities

Functionalities	Old	New
User Registration	Yes	Yes
Rides Registration	Yes	Yes
Rides from friends	Yes	Yes
Rides from user	Yes	Yes
Request of rides	Yes	Yes
Confirmation and request messages	Yes	Yes
Public Profile	Yes	Yes
Reputation	Yes	Yes
Ridematching	No	Yes
Groups	No	Yes

ride and choose that the system find similar users automatically, so, the system will call the ridematching algorithm which will generate cluster based on the history of request or through the present requests. The ridematching algorithm may find users and it will show a list of result with users who can form groups. The another way to create a group is looking by users through the search.

2.2.3 Functional requirements

We illustrate two of the functional requirements of the system by means of activity diagrams (ERIKSSON; PENKER, 2000). The diagram of Figure 4 shows how to get rides suggestions through the ridematching, as explained below:

1. The user must register or ask rides.
2. The system will call ridematching algorithm that will look for similar user's trajectory.
3. After ridematching, a list of users with similar trajectory will appear if algorithm found any results. On other hand, an alert message will appear to inform unsuccessful operation, in other words, ridematching didn't find users with similar trajectory and a list of random rides will appear.

Grouping of users may be created through ridemacthing algorithm or manually. The diagram of Figure 5 shows how create groups:

1. User must register a schedule with destination and departure points.
2. User can choose how find others users, for example, automatically or not. First option enable the application to search for users as we have already said above. The another option, the user will search by friends that ask or offer rides with similar trajectory.

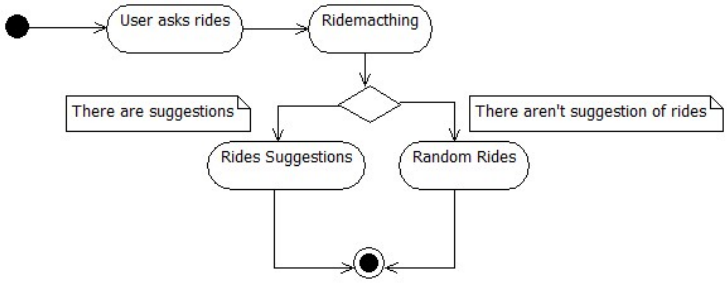


Figure 4: Activity diagram of ridematching operation.

3. After find similar users, the application will present a list of them.

4. The user can invite similar users to form a group.

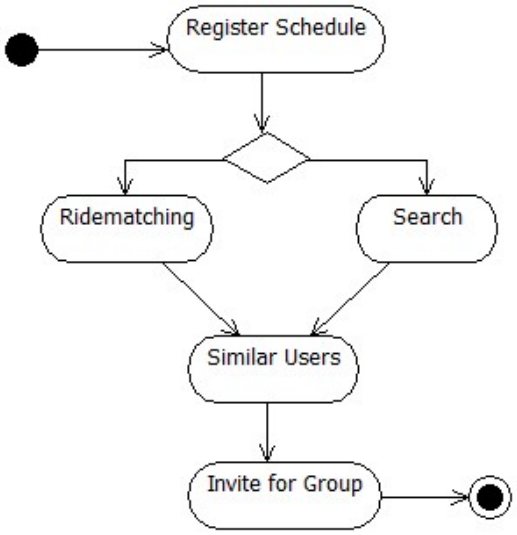


Figure 5: Activity diagram of grouping.

2.3 Method

Ridematching is a method to match similar rides. Similar rides, normally, is considered when users have similar trajectories and users have similar departure time. The ridematching approach used in GO!Caronas is defined by (CRUZ; MACEDO; GUIMARãES, 2015). The Figure 16 shows how the approach works.

The approach is divided into three steps: (i) trajectory discretization, (ii) temporal filter and (iii) clustering. Trajectory discretization is used to reduce a quantity of trajectory points. Considering that database C is set of user's trajectory.

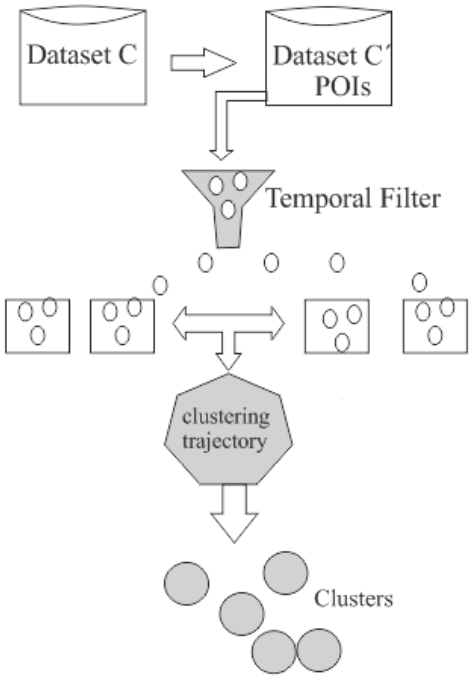


Figure 6: The method

Trajectory is represeted by points and each points is formed by triple $p_i = (lat, lng.timestamp)$. The discretization consists in computing a subset of points that is representative of the trajectory. The representative points are called POIs. The result of discretization enables the creation of the dataset $C^{'}$ which has only discrete trajectories according to the Figure 16.

A temporal filter is established so that only the set of users that have trajectories with similar departure and destination times are eligible for the processing pipeline to avoid processing waste.

Clustering is used to grouping similar users's trajectory. According to work (CRUZ; MACEDO; GUIMARãES, 2015), cluster algorithm Optics (ANKERST et al., 1999) was adapted to group users with similar trajectories. This algorithm uses the similarity function defined in (CRUZ; MACEDO; GUIMARãES, 2015). The function takes into account departure and destination points of passengers how is shown on Figure 19.

According to Figure 19, $Tr(d)$ belongs to driver user and $Tr(a)$ belongs to passenger. The main idea is to compare destination point of the passenger with all points of driver's trajectory and in the same way compare destination point. Those points are compared with all trajectory's points of driver in order to verify if passenger's trajectory and driver's trajectory are similar. The Figure 8 shows a real result which was reached by approach presented. The Figure 8 presents two similar trajectories that belong to the same cluster and represents two distinct users.

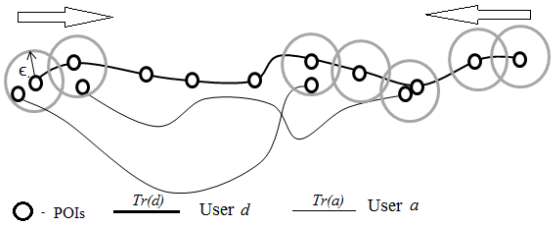


Figure 7: Approach used to define the similarity between two trajectories.

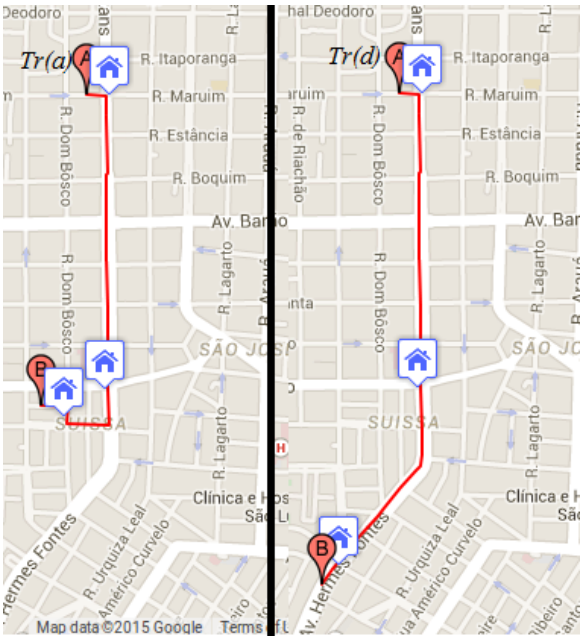


Figure 8: Real example of ridematching algorithm.

2.4 Results

A Beta version of the GO!Caronas is available [4]. Figures 9 and 10 show some screenshots of the system. These pictures show just the new functionalities as a group of rides and rides which are automatically found by ridematching approach. The Figure 9 shows the interface of application to register a ride by group.

The Figure 10 shows application's result related to a request done by user who has similar trajectory with another user.

We have reevaluated the system related with demand for carpooling service and evaluated the system through of the profiling approach. In the first case, we seek for the needs of potential users in regards to the services the system should provide. This evaluation compares results obtained, in 2013, with actual results.

In the second case, the system of ridematching has undergone automated profiling

[4] <http://kb.erickmendonca.com.br:8000/go_caronas/login/>

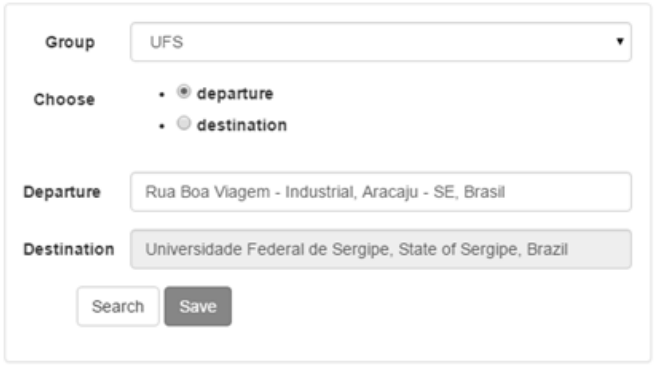


Figure 9: Screenshot of GO!Caronas - Register a ride in a group of ride.

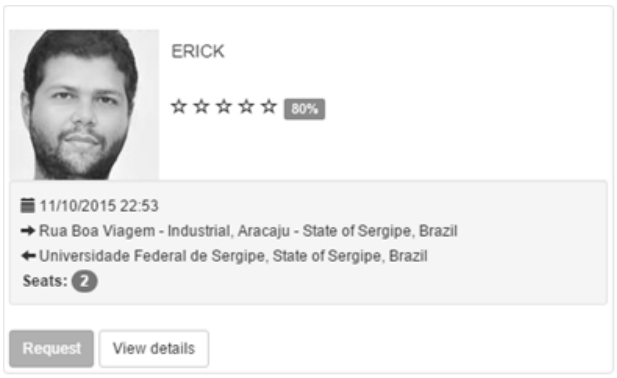


Figure 10: Screenshot of GO!Caronas - Example of result of ridematching.

analysis to verify which function, classes or library has been a bottleneck or the frequency and duration of functions call.

2.4.1 Prospecting the need for ridesharing

We applied a questionnaire [4] in order to find out what potential users think about and hope to get with a ridesharing on-line service and compared the current results with results obtained in (MATOS et al., 2014). More than 300 volunteers of Brazil were asked to respond the questions. The questionnaire was done and distributed through the Google Forms[5]. The Figure 11 presents the age of people who answered the questionnaire.

Three first questions allows us to observe important aspects. The first question refers to the usual number of occupants in the car the user uses to daily work. The result is shown in the graphics of Figure 12. We can observe that the result is almost equal, in other words, great part of respondents use the car alone or with no more than one passenger. Both graphics of Figure 12 show that there are not big changes in the results.

[4] <https://docs.google.com/forms/d/1XHGcHbfL7C0vGmkcEurybc7r0we_1ccebMpRrIRt7uk/viewform>

[5] <https://docs.google.com/forms>

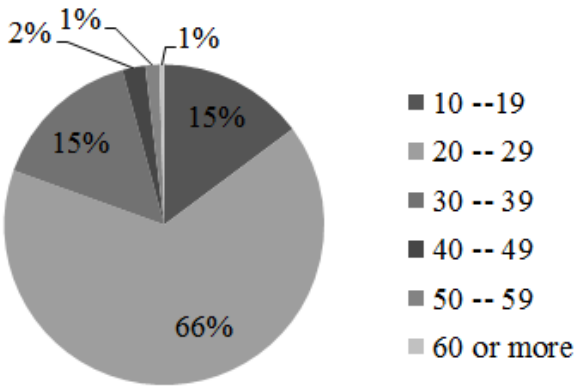


Figure 11: The range age of people.

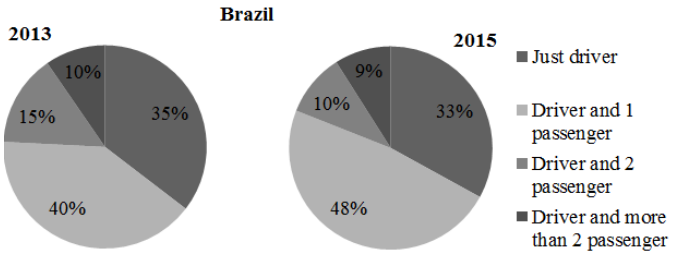


Figure 12: Number of occupants in cars

The sencond question, we analyze the importance of cultural concerns to get and provide rides Figure 13.

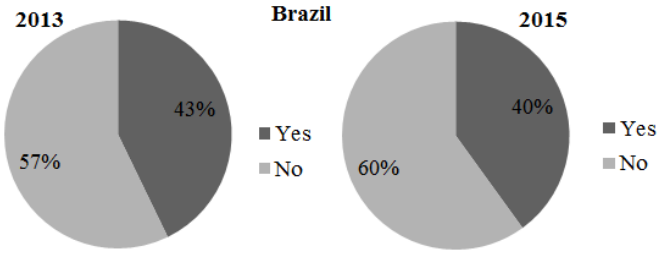


Figure 13: Cultural to deliver or get rides

Finally, in the third question, we analyze if the group functionality can encourage people use the system according to Figure 14.

2.4.2 Software profiling

The library cProfile [6] has been used to perform the profiling analysis. The analysis is done specifically in ridematching approach, because ridematching requires a lot of

[6] https://docs.python.org/2/library/profile.html

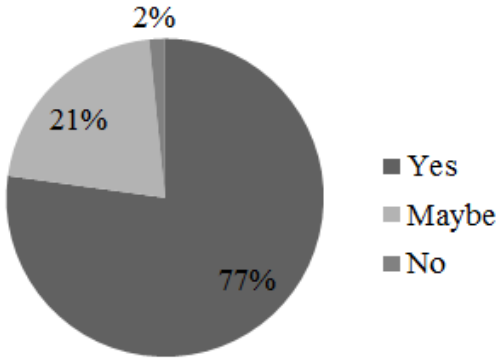


Figure 14: Form group may encourage user to get or offer ride.

computation to find clusters with users with similar trajectories.

The analysis cover a set of features such as: a number of call of functions $NumofCall$, the total time spent by functions or operations, the cumulative time spent by the functions $CTime$ etc. Besides, we checked the average of time that the method spends to run 500 trajectory of users was 25.019 seconds. The Table 8 shows three functions which are very expensive. Those functions are going to be improved.

Table 3: Profiling ridematching analysis.

Function	Num of Call	Total Time	Cum Time
math.cos	13317696	1.846	1.846
distance	1902528	16.182	22.297
neighbors	408	0.150	24.223

The Table 8 shows that function *math.cos* has been called a lot of times, but the time expensive spent by the function is less than 10% of *neighbors* function. The neighbors function is used by Optics algorithm and has fundamental importance.

2.5 Conclusion

This work shows the extension of the system GO!, an on-line social network and tool for ridesharing. The main goal of the extension is to enable that the system find ride in real time. GO!Caronas has been built with purpose to increase the possibility of people find rides. As a result, the user may not only sharing rides manually but also is able to receive a list of users which have similar trips. The second extension supports the functionality of group formation that enable users form groups with fixed trips. The rationale is that the user would have a greater incentive to offer rides and as a consequence the society awareness of more rational usage of private transport.

The questionnaire has shown that personal automotive continue underused and people need to stimulate the habit of getting and providing rides. We can also conclude that people tend to use systems that may encourage such good habit although security concerns are considerable.

GO!Caronas is the second step towards building a platform that encourage ridesharing. As future work, we intend to provide integration with a new approach of ridematching which take into account trajectory and users profile to find similar rides. Finally, we intend to work on an improvement to enable personalized recommendation of rides and friends.

3 Measuring the relevance of the trajectory matching and the profile matching on the context of carpooling computational systems.

In this chapter, we propose to extend the method propounded by (CRUZ; MACEDO; GUIMARãES, 2015). We suggest an innovative approach which permits to generate clusters in the context of carpooling based on user's profile and/or trajectory, in other words, our approach can produce three final results such as, clusters of users with similar trajectories, users with similar profile and user with similar trajectory and profile. We also define a formalization of terms related to ridesharing context, more specifically in carpooling context. We used K-means (MACQUEEN et al., 1967) to group users with similar profiles and employed particular part of clustering ensemble approach, based on voting (MENG; TONG; WANG, 2011), to combine clusters of user's trajectory (CRUZ; MACEDO; GUIMARãES, 2015) in order to generate final clusters which have users who are similar in profile and trajectory.

The rest of this chapter is organized as follows. Section 5.1 reviews some works that use some kind of profile matching in the context of carpooling. Section 5.2 reviews some ensemble clustering approach. In section 5.3, we describe the global approach to generate user's clusters with similar profiles and trajectories. In this section, we explained which part of ensemble learning based voting is used. In section 5.4, Experiments and results are presented. Finally, we conclude the work in section 5.5.

3.1 Profile Matching

The social distance is characteristic that hasn't been explored in the context of carpooling by researchers. Information about similarity among people that have interest in share any ride can be very important to encourage users decide to accept or deny a request of a ride. As aforementioned, characteristics that influence social distance such as gender, age, smoke can be a factor of security or safety.

(YAN; CHEN, 2011) proposed to employ a time-space network flow technique to develop a model that can be used to solve problems of carpooling with pre-matching information. This pre-matching information uses some characteristics such as if user smokes or doesn't and the gender to define final riders. According to the work(YAN;

CHEN, 2011), carpool group (CG) is defined as CNG that is a CG that does not provide a vehicle. CNG requests are defined with characteristics which are classified into four types depending on gender and smoking status: non-smoking female, smoking female, non-smoking male and smoking male. The work defines CNG types of requests: (1) riding with non-smoking females; (2) riding with females; (3)riding with non-smokers; (4) non-requester. In accordance with work, those types of CNG requests are restrictions or some of type of filter to compose the network flow technique in order to find final ridematching. The Figure 15 shows a simple example of the pre-matching information. Some restrictions can be given, for example, smokers requesting a ride with non-smokers do not match etc.

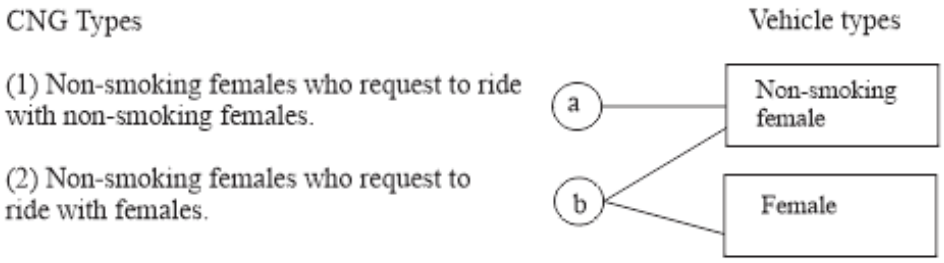


Figure 15: Relation between CNG requests and vehicle types.

(HE; HWANG; LI, 2014a) proposed a social distance that measure the relationship between riders and passengers. But, they simply measure the distance between rider's home and office. (DOYTSHER; GALON; KANZA, 2011) presents a graph model for socio-spatial network that stores information on frequently traveled routes. That work considers a social network which is a graph whose nodes represent real-world people and whose edges represent relationship between people and proposed a query language that consists of graph traversal operation to expedite the formulation of queries over the network. The paper suggests that each user of social network can be represented by personal properties like hobbies, name etc.

Our work proposes to use some characteristics of users that can be considered common in social network such as: gender, age, hobbies and reputation. The former attribute is very important to the context of carpooling, because through the reputation, users can qualify other users. We propose a different approach in context of carpooling. We apply K-means method to find similar users according to any attributes. In this work, specifically, we used attributes that were chosen according to (FURUHATA et al., 2013), (YAN; CHEN, 2011), and others were chosen according to a questionnaire applied.

3.2 Clustering Ensemble

Clustering ensemble is focused on combining strengths of many individual clustering algorithms (GHAEMI et al., 2009). The focus of clustering ensemble is to permit that final clusters can get better results, because it can go beyond what single clustering algorithm in several aspects such as robustness, novelty, stability and scalability (GHAEMI et al., 2009). Furthermore, it is possible to combine clusters of different datasets, clusters with different features of the same dataset and clusters of different algorithms.

Clustering, normally, require the definition of similarity measure between features which is a tough task without prior knowledge about cluster shape (FRED; JAIN, 2002). According to literature, cluster ensemble has brought improvements in clusters results and this technique has been used in real world application as video retrieval (CHANG et al., 2008), cluster analysis (FRED; JAIN, 2002) and feature selection (HONG et al., 2008).

In general, there are two stage on algorithms to use ensemble clustering: (i) store results of independent clusters that belongs to the same or different cluster algorithm, (ii) the consensus function is used to find a final partition. Consensus function define the way that different clusters can be combined. There are some clustering combination approach as voting, co-association based function, hypergraph partitioning, finite mixture model etc.

(MENG; TONG; WANG, 2011) proposes a new clustering ensemble algorithm, based on voting, and presents a correlation to represent similarity of clusters. This work proposes a consensus function called RELABEL which uses k-means to produce cluster members and introduces correlation to unify cluster labels. (IQBAL; MOH'D; KHAN, 2012) proposes engaging supervision in the clustering ensemble procedure to get more enhancements on the clustering results.

(STREHL; GHOSH, 2003) proposes three kind of consensus function:the first induces similarity measure from the portioning and it is called Cluster-based Similarity Partitioning Algorithm (CSPA) which consider the relationship between objects in the same cluster and uses it to establish a measure of pairwise similarity; the second function is called HyperGraph Partitioning Algorithm (HPGA) and the third algorithm is called Meta-Clustering Algorithm (MCLA). The HPGA is based on hypergraph on a graph whose vertices correspond to the objects belonging to the same clusters.

3.3 Method

Our approach extends clustering trajectory proposed by (CRUZ; MACEDO; GUIMARãES, 2015) with K-means algorithm as follows. Given a set of users $U = \{S_1, S_2, ..., S_n\}$, where each S_i is represented by tuple formed by a user's trajectory Tr_i and user's profile P_i, Optics* generates a set of cluster $A = \{C_1, C_2, ..., C_n\}$, where each

$C_i = \{Tr_1, Tr_2, ..., Tr_n\}$ represents a set of user's trajectory and it has at least one trajectory from a user called driver d that provides a ride. In sequence, K-means generates a set of cluster $B = \{X_1, X_2, ..., X_n\}$, where each $X_i = \{P_1, P_2, ..., P_n\}$ represents a set of user's profiles. Finally, a ensemble approach is used to combine the set of cluster A and B to generate a final set of clusters R which has clusters of users that have similar profile and trajectory.

Definition 1 *A driver is a user who shares a vehicle with passengers and has similar trajectory with all passengers. Tr(d) is a trajectory that belongs to the driver.*

In this work, $Tr(d) \sim U = \{Tr(a_1), (a_2), ..., Tr(a_n)\}$ means that driver's trajectory and passengers' trajectory are similar, for example, $dist(Tr(d), Tr(a)) \leq r$, where $dist()$ is some distance function and r is a boundary distance.

Definition 2 *A vehicle is defined as any means which someone travels such as car, motocycle etc. Here, a vehicle is represeted by V, where V(d) is a vehicle that belongs to the driver d.*

Definition 3 *A passenger is a user who shares a vehicle with a driver and has similar trajectory with a driver. Tr(a) is a trajectory that belongs to a passenger a.*

In this work, $Tr(a) \simeq Tr(d)$ means that driver's trajectory and passenger's trajectory are similar.

Definition 4 *Ride is defined as a way to share a private vehicle space among people with similar trajectory and interests. A ride is represented by $R = (V(d), d, Tr(d), A)$, where V(d) is driver's vehicle, d is a driver, Tr(d) is a driver's trajectory and A is set of passengers.*

Figure 16 depicts the whole method. Note that clustering process takes into account special distances among trajectories and social distance among user's profile.

The method is divided into five steps: (i) discretization of user's trajectory, (ii) temporal filter, (iii) Optics clustering, (iv) K-means clustering and (v) relabel and intersection clusters. The three first steps are developed by (CRUZ; MACEDO; GUIMARãES, 2015), consequently, they are presented briefly. Next subsections detail each of them.

3.3.1 Trajectory's discretization

Consider U a set of user's trajectories that have a large number of points. Many of such points are redundant due to the short time interval in which they are obtained. Through the RotaFacil (TELLES; GUIMARÃES; MACEDO, 2012), (TELES et al., 2013),

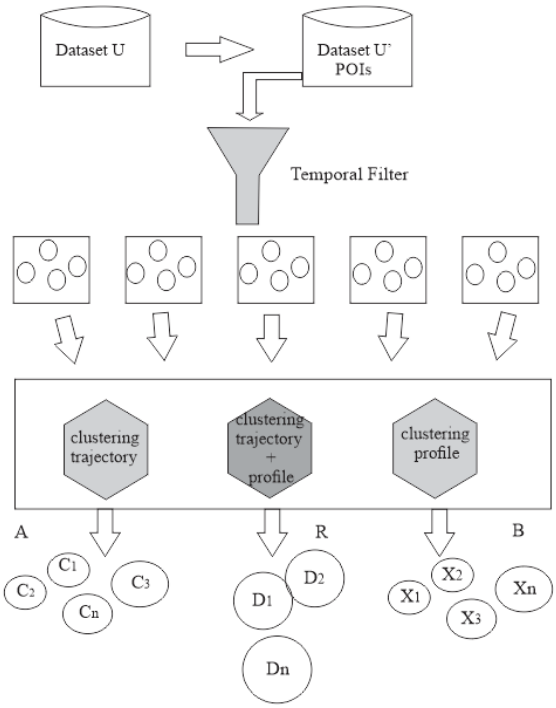


Figure 16: Method

trajectory's discretiazation is computed and a new subset U' is generated. A subset is composed by a trajectory which is formed only by representative points (POIs). The Figure 17 presents a trajectory discretized with some POIs.

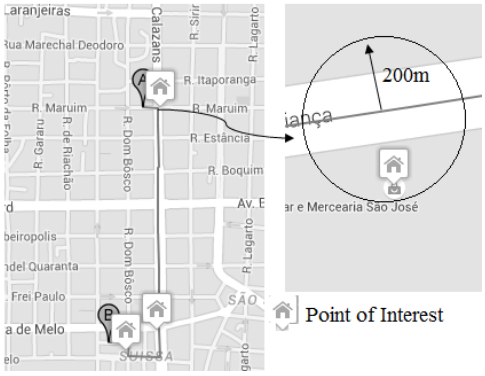


Figure 17: POIs within a circumference for a given radius threshold.

3.3.2 Temporal filter

A temporal filter is established so that only the set of trajectories with similar departure and destination times are eligible for the processing pipeline to avoid processing waste. Indeed, it is pointless clustering together $Tr(d)$ and $Tr(a)$ if departure and/or destination times of users d and a are rather distinct, even though $Tr(d) \simeq Tr(a)$.

Considering t the time of such a ride offering and x is a bound of time informed by user, the width of the filter is the interval $[t - x, t + x]$ (Figure 18).

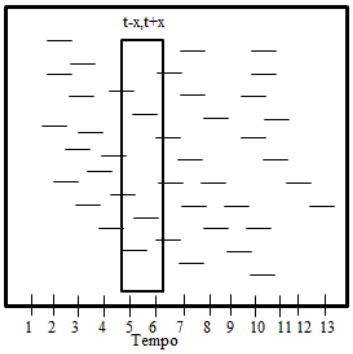


Figure 18: Temporal filter to trajectory clustering.

3.3.3 Optics clustering

The clustering trajectory is realized by Optics* algorithm which is the Optics algorithm (ANKERST et al., 1999) adapted by (CRUZ; MACEDO; GUIMARãES, 2015). We used the Haversine function to calculate the distance between two points and used a similarity algorithm defined by (CRUZ; MACEDO; GUIMARãES, 2015) to calculate the similarity between two trajectories.

The similarity algorithm has its behavior presented by the Figure 19 where $Tr(a)$ belongs to a passenger a who wishes to get a ride and $Tr(d)$ belongs to driver. The similarity just takes into account origin and destination points of the trajectory of the passenger.

Definition 5 *Origin is defined as the first point* p_1 *of each trajectory, where* $p_1 \in Tr$.

Definition 6 *Destination is defined as the last point* p_n *of each trajectory, where* $p_n \in Tr$.

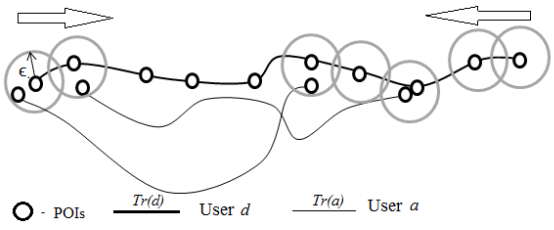


Figure 19: Approach used to define the similarity between two trajectories.

3.3.4 K-means clustering

For K-means algorithm, a number of clusters k must be informed as a prior knowledge. K information is got through of the number of clusters generated by Optics*. According to the context of our work, it is interesting to have the same number of the clusters as a result of Optics* and K-means, because once we find out partitions with the same number of clusters, the relabel algorithm can be applied more easily.

To calculate the similarity between profile of users with K-means, it was used cosine function similarity (THEODORIDIS et al., 2010) between two users $sim(P_i, P_j)$.

$$sim = \cos(\theta) = \frac{P_i.P_j}{\|P_i\|, \|P_j\|} \tag{3.1}$$

The result of similarity lies in the interval [-1,1], where -1 indicates that user's profiles are opposite, 1 means the same profile and 0 shows that user's profiles are independents or $\theta = 90°$.

3.3.5 Relabel and Intersection

In clusters, there are not labels to identify and distinguish clusters like occur in supervisioned algorithms. So, combining clusters of the different algorithms demands some approach to identify similar clusters. Relabel strategy is a way to align clusters considered similar as a show in Figure 20.

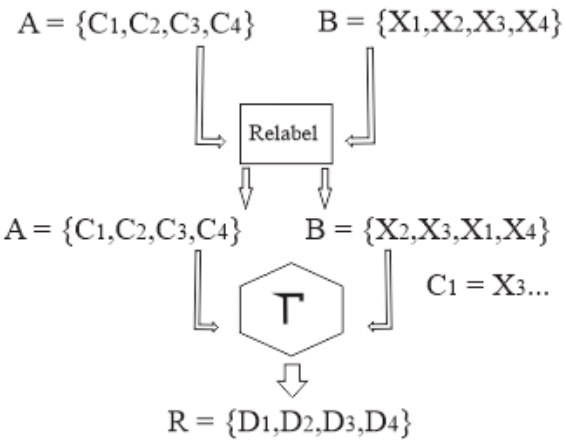


Figure 20: Approach used to relabel and realing clusters.

The partition A and B found out by Optics* and K-means are processed by Hungarian algorithm (KUHN, 1955) to align and relabel the partitions in order to verify which clusters have more users in common, for example, consider that the cluster C_1 has 4 users with similar trajectories and the cluster X_2 has 10 users with similar profiles, if we

take into account all combinations between A and B, X_2 and C_1 are clusters which have more users in common.

Figure 21 shows the basic process of relabel. The columns represent the base partitions and rows represent the users. The permutation is used to align the most similar clusters. Users that belong to similar clusters will make part of final clusters. This work uses trajectory partition as a reference partition which is used with base to align other partitions. As Figure 21 and 20 show, voting approach is not used totally, because in the context of our work we did not have the necessity to vote to generate final clusters with just two partitions.

Definition 7 *Trajectory is a sequence of multi-dimensional points. These points are discrete and finite and they are represented by* $Tr = \{p_1, p_2, p_3, ..., p_n\}$. *Here* p *is a 3-dimensional point that is formed by latitude, longitude and timestamp,*$p = \{lat, lng, t\}$.

	A	B
S_1	1	A
S_2	1	A
S_3	2	C
S_4	3	D
S_5	2	C
S_6	3	D

	A	B	FC
S_1	1	1	D_1
S_2	1	1	D_1
S_3	2	2	D_2
S_4	3	3	D_3
S_5	2	2	D_2
S_6	3	3	D_3

Figure 21: Basic relabel and assembly final clusters.

The consensus functions is represent by τ on Figure 20. The function considers the intersection between clusters as shown in 2.

$$\tau(C_i, X_j) = C_i \cap X_j \tag{3.2}$$

The intersection function is used as a consensus function because the final partition D results in two partition A and B. The final partition must have clusters that have users who are similar in trajectory and profile.

3.4 Experiments

We have performed four experiments in order to prove the feasibility of the approach to ridematching with trajectory and profile in the context of carpooling. The first two experiments was rerun according to (CRUZ; MACEDO; GUIMARãES, 2015) and show

results between trajectories clusters generated by Optics*. The third experiment shows the results of clusters of users profile generated with Optics and K-means. Finally, the fourth experiment show results of clusters with users that have similarity between trajectory and profile.

3.4.1 Datasets

We use three different datasets. The first dataset consists of actual trajectories collected from users of the Go!Track[1] app (GOOGLE, 2013). GO!Track continuously collects GPS points of the trajectories people are taking while in theirs cars. This set currently contains around 40 trajectories with more than 5,317 points. Second dataset consists of 500 trajectories, artificially generated by RotaFacil. The third dataset consists of 500 registers of profile attributes artificially generated.

3.4.2 Experimentation setup

Discretization applied to the first dataset has reduced the original 5,317 points to 394, considering the radius of 200 meters in RotaFacil.

In order to generate the artificial trajectories, $n = 23$ addresses were randomly chosen. Every address consisted of a origin point and a destination point. The number of different trajectories was then $n^2 - n$.

For first two experiments, we have varied ε Optics* parameter with 100m, 200m, 300m and $MinPts$ parameter with 2 and 3. We have assumed that 100 to 300 m are reasonable distance limits for a user who desires a ride to move towards the destination point of a offering ride. For the cluster extraction algorithm, the values were set to 50, 150 to parameter ε'.

For the last two experiments, five attributes were chosen to characterize users profile:(i) gender (binary), (ii) reputation (discrete), (iii) age (discrete), (iv) smoking (binary), (v) music (discrete). All the attributes were normalized in order to vary between 0 and 1.

3.4.3 Evaluation metrics

Davies-Boundin index (DBI) (THEODORIDIS et al., 2010) was used to evaluate the clustering task. Equation 3 defines the DB value:

$$DBI = \frac{1}{n}\sum_{i=1}^{n} max_{i \neq j}\left(\frac{\alpha_i + \alpha_j}{d(c_i, c_j)}\right) \tag{3.3}$$

[1] https://play.google.com/store/apps/details?id=com.go.router

where n is the number of clusters, c_i and c_j is the centroid of each cluster. The α_i and α_j are the similarity measures for clusters c_i and c_j.

The values generated by Equation 3 reflect how similar the elements of the same cluster are, as well as the dissimilarity among distinct clusters. Smaller DBI values are better.

3.4.4 Experimentation results

Table 14 shows the results of the first experiment. The number of clusters *NC* is zero when $MinPts$ is 3 according to experiment presented by (CRUZ; MACEDO; GUIMARãES, 2015).

Zero values occurred in Table 14, because the number of neighbors of the user trajectories was less than 3. This show that the real base have not many similar trajectories.

Table 4: Results for the dataset with actual trajectories.

*Optics**	ε	$MinPts$	$\varepsilon^{'}$	NC	DBI
	200	2	150	3	0.2885
	200	3	150	0	0
	300	2	250	3	0.2885
	300	3	250	0	0

Table 5 shows the results of second experiment. The DBI values among three algorithms are similar.

Table 5: Results with artificially generated dataset

*Optics**	ε	$MinPts$	$\varepsilon^{'}$	NC	DBI
	100	2	50	39	0.7868
	200	2	150	54	0.8860
	200	3	150	34	0.7235
	250	2	150	62	0.8968
	250	3	150	36	1.0164
	300	2	100	48	0.7946

ε directly influences clusters' size according to Table 5. Any ε that is "big" enough will produce good results. Unlikely, small ε will produce a lot of objects with *reachability-distance* value equal to *undefined*. In this work, as well as in the work (CRUZ; MACEDO; GUIMARãES, 2015) neither method was used to deduce the perfect ε.

The third experiment is showed by Table 6. The experiment presents a comparison of clusters results that take into account user's profile. We just show some results and we can verify that K-means has better results when a number of clusters grow up.

Table 6: Results with artificially generated dataset

Optics	ε	$MinPts$	$\varepsilon^{'}$	NC	DBI
	0.5	4	0.1	4	1.5949
	0.5	3	0.1	15	2.741857
	0.5	2	0.1	103	3.35144
K-Means				k	
	0	0	0	4	2.16712
	0	0	0	15	2.5023
	0	0	0	103	1.467086

The results of Table 6 help us to choose K-means as algorithm to generate profile clusters, consequently, K-means was used in the fourth experiment.

The Table 7 shows results of ensemble learning approach to get clusters that have users with similar trajectories and profiles. So, the results presented were obtained through of matching done with trajectory clusters generated by Optics* and profile clusters got with K-means. The Table 7 shows a number of final clusters (NFC), Davies-Boulding Index related to trajectory (DBIT) and Davies-Boulding Index related to profile (DBIP).

Table 7: Results with artificially generated dataset

Ensemble	NC	NFC	$DBIT$	$DBIP$
	68	10	0.608994	1.18678
	44	11	0.5095	0.99538
	66	11	0.6619119	1.50945
	69	8	0.377242	3.35144

The Figure 22 presents another viewpoint of results shown in Table 7. It shows the results of the Optics* and K-means taking into account the number of clusters (NC) before the ensemble learning approach by DBI metrics. We can verify that DBI metrics is better when a number of clusters is larger. These results happened, probably, because the artificial dataset that was used in this experiment, does not have many user's trajectory with the similarity less than 200 meters or user's profile are not so similar according to values used in this work.

3.4.4.1 Analysis of Complexity

We have performed some analysis to verify the cost of complexity of our approach. In order to get some results, in the first moment, we used the software profiling method

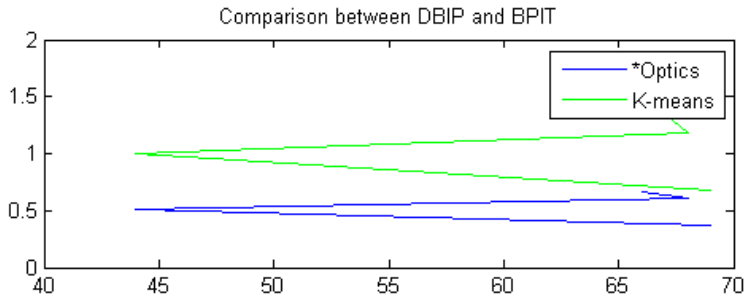


Figure 22: Comparison between DBIP and DBIT by NC.

which is a form of dynamic program analysis. In the second moment, we calculate an estimate of the complexity according to big O notation.

The software profiling was done through the library cProfile [2]. The profiling was used with the purpose of determining which part of the method must be optimized. The analysis cover a set of features such as: a number of call of functions $NumofCall$, the total time spent by functions or operations, the cumulative time spent by the functions $CTime$ etc. Besides, we can verify the quantity of time that all the method proposed spends. The Table 8 shows four functions which are most expensive. The results of Table 8 were obtained with the following setup: $\varepsilon = 100$, $MinPts = 2$, $\varepsilon^{'} = 150$, $k = 54$.

Table 8: Profiling method analysis.

Function	Num of Call	Total Time	Cum Time
math.cos	13317696	1.846	1.846
distance	1902528	16.182	22.297
neighbors	408	0.150	24.223
mean	40	0.04	0.0053

The Table 8 shows that function *math.cos* has been called a lot of times, but the time spent by the function is less than 10% of neighbors function. The neighbors functions is used by Optics algorithm and has fundamental importance. The neighbor function is a bottleneck belongs to Optics algorithm, because it consults all trajectories when it is called. According to (ANKERST et al., 1999), a index structure like tree-based spatial index can be used and consequently decrease the overall run-time.

The first analysis of method complexity was done. According to (ANKERST et al., 1999), the Optics* algorithm has a overall run-time of $O(n^2 . \lg n)$ considering a spatial index and similarity algorithm. Besides, K-means algorithm has overall run-time of $O(n^{dk+1} . \lg n)$ where d is a dimension and k a number of clusters. The ensemble approach used has a

[2] https://docs.python.org/2/library/profile.html

overall run-time of $O(K^3)$ considering that Hungarian is been employed. The run-time of ensemble approach can be considered constant, because K is a number of partition that is fixed in 2. Then, the global run-time is $O(n^2.\lg n + n^{dk+1}.\lg n)$.

3.4.4.2 Prospecting the attributes for carpooling

We applied a questionnaire [3] in order to find out what potential types of attribute are considered important to users to get or reject a ride. More than 320 volunteers of Brazil were asked to respond the question. The Figure 36 shows results. We can observe that attributes like gender, smoking and age are considered more relevant as suggested by (FURUHATA et al., 2013).

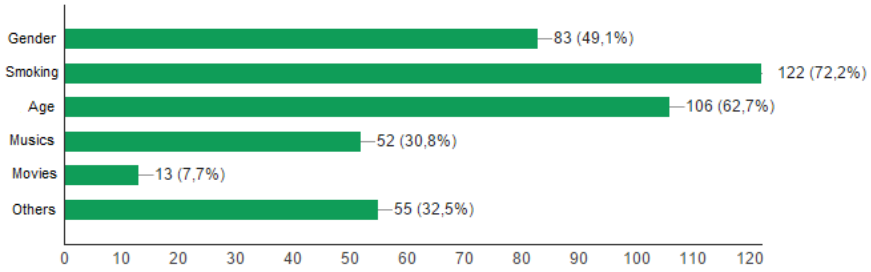


Figure 23: Relevant attributes of profile according to questionnaire.

3.5 Conclusion

Encouraging carpooling is an important effort towards the reduction of in-transit vehicles. Although there is some concerned research initiative and even some related software, they do not appropriately treat carpooling context specificity. In this paper, we have proposed extends a method developed by (CRUZ; MACEDO; GUIMARãES, 2015) to deal with some of these specificity such as find out groups of users that have similar profile and trajectory and consequently discover potential carpooling opportunities. Furthermore, clustering users trajectory and clustering users profile are results that can be got separately according to final interest of who desire to utilize the proposed approach.

Clustering results and corresponding *Davies-Bouldin Index* values obtained from a dataset of actual trajectories collected pervasively have shown the feasibility of the proposal. Furthermore, the initial method analysis of run-time shows that the approach has high complexity. However, if considering that the problem of find out similarity among trajectories of users is a special case of the so-called pickup and delivery problem which is NP-Complete (AGATZ et al., 2012), the results of analysis show the feasibility.

[3] https://docs.google.com/forms

We are currently working with experiments that considers that attributes profile have weights in order to permit that users can define which attributes are considered more important. We also intend to develop a carpooling recommend service so it could be integrated to some carpooling software, such as the GO!Caronas (MATOS et al., 2014), (HENDRIK TEXEIRA MACEDO., 2014). Besides, we want to compare our approach to generate ridematching and profile matching with the approach used by (CARVALHO; MACEDO, 2013) which uses coalition structure to provide proper group's formation.

4 An open urban mobility dataset

This chapter contributes to scientific community by conceiving an open dataset for real traffic data. Trajectories of cars and buses have been collected in empirically defined time-stamps. Dataset attributes are: time, distance, speed, bus line, condition of weather, quality of the travel,latitude, longitude. The so-called GO! Track is the mobile application underlying the collection routine equipping volunteers' devices. We have made GO! Track freely available as well. In order to show the dataset usefulness, we provide examples of machine learning tasks.

The remainder of this chapter is organized as follows. Section 6.1 present some works concerned to providing urban datasets. Section 6.2 presents GO!Track application and dataset, which is properly evaluated in section 6.3. Examples of dataset usage by some machine learning algorithms are also shown in section 6.3. Section 6.4 shows the dataset usage for some user-end applications. Finally, we conclude the work in section 6.5.

4.1 Related Works

Mining trajectory and trajectory similarity have been focus of recent academic research, although there are not much available datasets with real data on urban mobility.

(ZHENG; XIE; MA, 2010) presents a collaborative social networking that aims to reasoning on the trajectories, locations and users in order to generate travel recommendations and the sharing of life experiences. (YAN et al., 2011) presents a framework that enable the annotation of trajectories for any kind of moving objects. (HERRERA et al., 2010) presents a traffic monitoring system based on smartphones provided by GPS. 100 vehicles, carrying phone, drove on a 10-miles stretch at California, for 8 hours. The authors argue that it is possible to provide accurate measurements of the velocity for the traffic flow with just 2-3% of phones that belong to the driver population. In all three works, trajectories points are collected by GPS devices but there isn't the purpose to release a well structured open dataset of trajectories freely available to researchers explore new methods, algorithms and approaches to solve urban mobility issues.

ChoroChornos.org [1] provides a dataset about trucks traffic with attributes like date, time, latitude, longitude, x and y for GGRS87 (Greek Geodetic Reference System 1987) reference system. Figure 24 shows a map of the track with 276 trajectory.

Data.rio [2] makes available a dataset of bus traffic. It has been an initiative of Rio

[1] http://chorochronos.datastories.org/?q=node/5
[2] http://data.rio/dataset/gps-de-onibus/resource/cfeb367c-c1c3-4fa7-b742-65c2c99d8d90

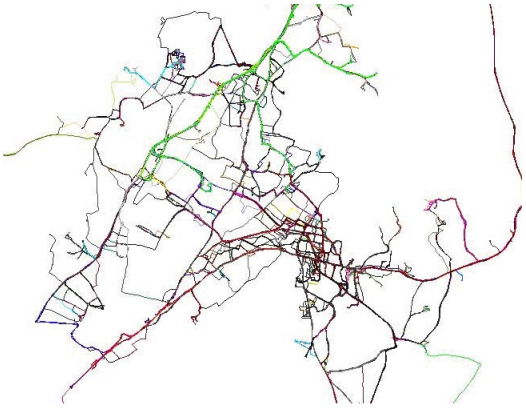


Figure 24: Plot of truck trajectory.

de Janeiro city and has some attributes such as, bus line, latitude and longitude, date, hour, and bus speed.

Archive.org [3] has made available a dataset about New York taxi trip. It has attributes such as pickup datetime, dropoff latitude, etc.

The Mobile Millennium [4] is a project developed by the California Center of Innovative Transportation (CCIT), the Nokia Research Center and the University of California (UC) at Berkeley. The project aimed to verify the potential use of mobile phone and navigation technologies to monitor real-time traffic flow. The project has created a dataset with track of GPS about traffic data.

In ShareMyRoute.com [5], people can share information about the grate outdoors. The application allows users to share their routes, but does't provide the option to download a dataset of the trajectories. It's possible make to download of individuals trajectory, but there isn't the option to get a log of trajectory as in ChoroChornos.org.

4.2 GO!Track

The works presented so far have in common the use of attributes focused on providing understanding on general dynamic behavior of urban mobility in cities. The proposal of GO!Track's dataset is to provide data specifically on car traffic flow and bus traffic flow. GO!Track is freely available [6] and the dataset is continuously being feed [7,8]. GO!Track allows users to track on two kind of transportation data: (i) private car and (ii) bus.

[3] https://archive.org/details/nycTaxiTripData2013
[4] http://traffic.berkeley.edu/project
[5] http://www.sharemyroutes.com/
[6] https://play.google.com/store/apps/details?id=com.go.router
[7] https://go-goproject.rhcloud.com/
[8] http://archive.ics.uci.edu/ml/datasets/GPS+Trajectories

4.2.1 Application interface

GO!Track app has some graphical user interfaces to enable easy access to underlying functionalities. Users may choose the type of vehicle 25 and then he/she may track, correspondingly (Figure 26).

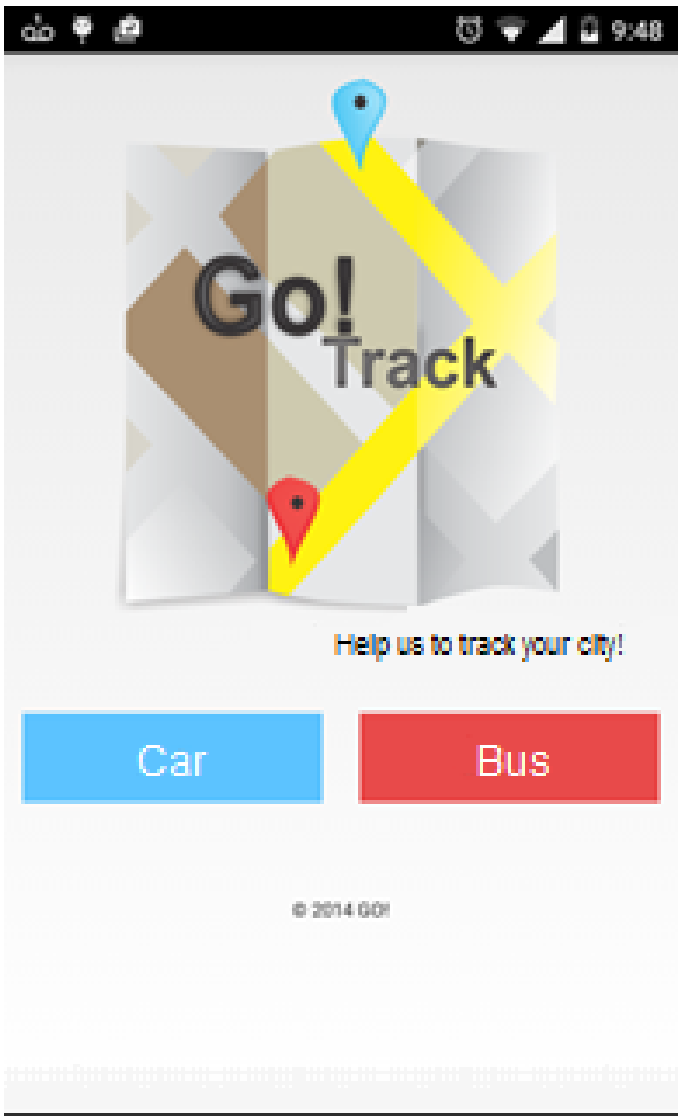


Figure 25: Choosing vehicle in GO Track!.

Once the journey is finished, users can evaluate it according to weather and traffic flow (Figure 27).

4.2.2 The dataset

The dataset consists of 163 routes, from 28 different mobile devices of different users. Each route consists of a set of points collected at an interval of 05 seconds for car traffic and 10 seconds for bus traffic. These values have defined empirically. All points were collected in the city of Aracaju/SE. Table 9 summaries the dataset.

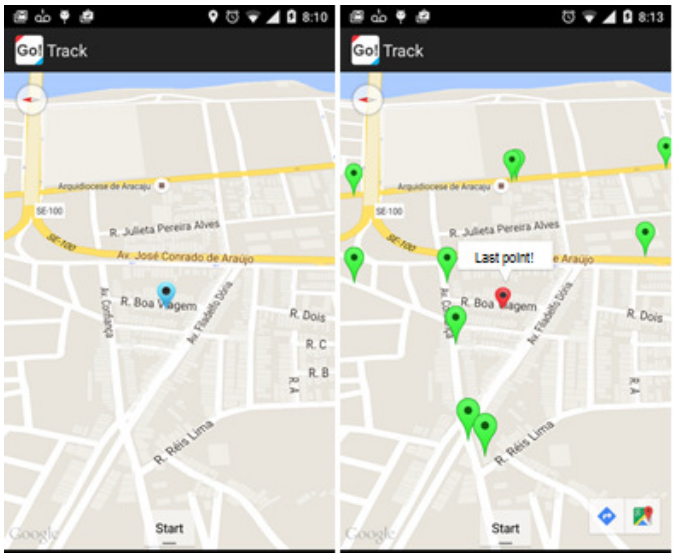


Figure 26: Tracking vehicles. Scattered green markers are bus stops.

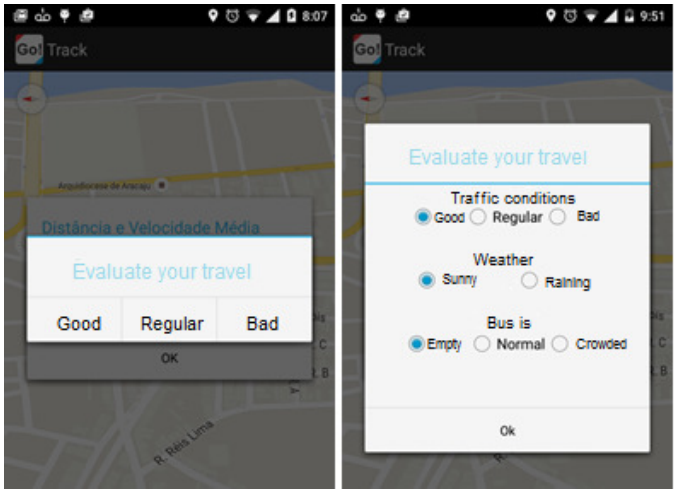


Figure 27: Evaluation of the journey.

Table 9: GO! Track dataset.

Measures	values
Period of collect	09/14 - 01/16
Number of routes	163
Number of different devices	28
Mean of points by route	111.08
Number of car routes	87
Number of bus routes	76
Number of points collected by cars	14011
Number of points collected by buses	4096
Distinct Address visited	428

The dataset is organized in two different tables (10 and 11). First table stores the

routes that have been collected and the second one stores the geographic points (latitude and longitude).

Table 10: Collected routes

Field	Description
id	unique key to identify each route.
id_android	an identifier for each device that was used to collect routes.
time	the duration of the pathway in minutes.
distance	distance of the route in kilometer.
speed	average speed during all pathway.
rating	the user evaluation about the traffic.
linha	information about the bus that does the pathway (available just in bus case).
car_or_bus	indicates if the route was collected by a car or a bus.
rating_wheather	indicates the conditions of the weather (available just in bus case).
rating_bus	indicates the quality of the travel (available just in bus case).

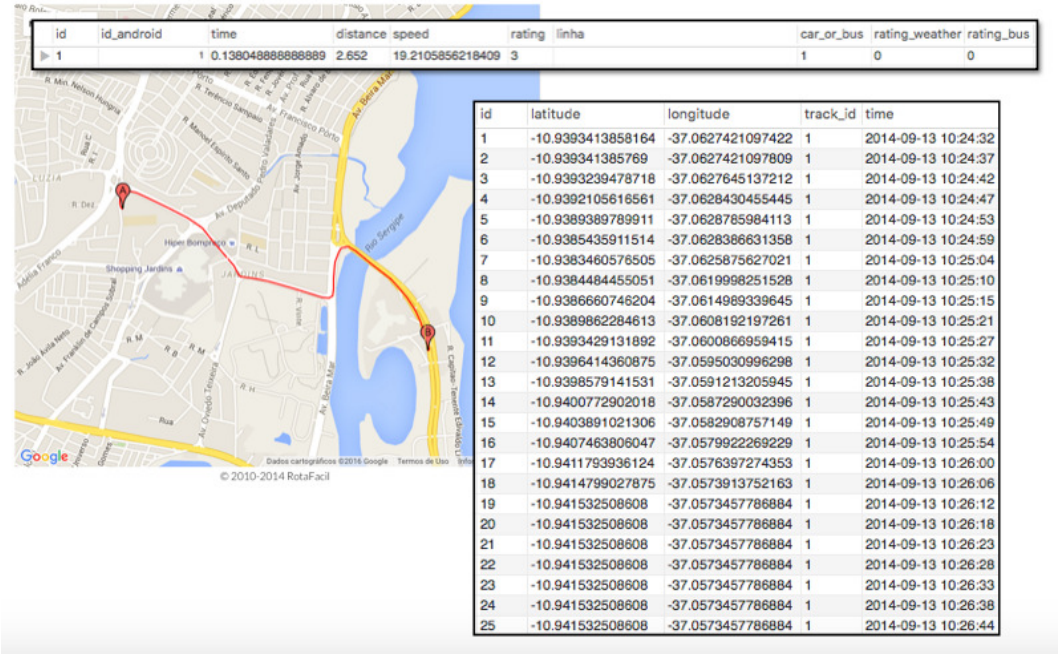
Table 11: Geographic points

Field	Description
id	unique key to identify each point
latitude	latitude from where the point is
longitude	longitude from where the point is
track_id	identify the route to which the point belongs
time	datetime when the point was collected

Figure 28 shows a screenshot of a route and its corresponding data.

The current version of the dataset covers an important set of streets and avenues of Aracaju city. We have plot all the dataset points on Aracaju city map (Figure 29). In addition, table 12 lists the top-20 most visited traffic roads by GO!Track users, according to the **number of routes** that actually used it. This was accomplished by the Google Geocode API. The column **number of points** shows the number of points presented in that traffic road, regardless of the route. We highlight the known main city traffic roads in respect to traffic density in peak hours.

Many practical applications use date-time data to predict traffic conditions or travelling time. Following graphs provide relations between the geographic points and the time. The graph of Figure 30 shows the relationship between the routes and date-time. The X-axis represents the 163 routes and the Y-axis the time bands. Each line in the



id	id_android	time	distance	speed	rating	linha	car_or_bus	rating_weather	rating_bus
1	1	0.138048888888889	2.652	19.2105856218409	3		1	0	0

id	latitude	longitude	track_id	time
1	-10.9393413858164	-37.0627421097422	1	2014-09-13 10:24:32
2	-10.939341385769	-37.0627421097809	1	2014-09-13 10:24:37
3	-10.9393239478718	-37.0627645137212	1	2014-09-13 10:24:42
4	-10.9392105616561	-37.0628430455445	1	2014-09-13 10:24:47
5	-10.9389389789911	-37.0628785984113	1	2014-09-13 10:24:53
6	-10.9385435911514	-37.0628386631358	1	2014-09-13 10:24:59
7	-10.9383460576505	-37.0625875627021	1	2014-09-13 10:25:04
8	-10.9384484455051	-37.0619998251528	1	2014-09-13 10:25:10
9	-10.9386660746204	-37.0614989339645	1	2014-09-13 10:25:15
10	-10.9389862284613	-37.0608192197261	1	2014-09-13 10:25:21
11	-10.9393429131892	-37.0600866959415	1	2014-09-13 10:25:27
12	-10.9396414360875	-37.0595030996298	1	2014-09-13 10:25:32
13	-10.9398579141531	-37.0591213205945	1	2014-09-13 10:25:38
14	-10.9400772902018	-37.0587290032396	1	2014-09-13 10:25:43
15	-10.9403891021306	-37.0582908757149	1	2014-09-13 10:25:49
16	-10.9407463806047	-37.0579922269229	1	2014-09-13 10:25:54
17	-10.9411793936124	-37.0576397274353	1	2014-09-13 10:26:00
18	-10.9414799027875	-37.0573913752163	1	2014-09-13 10:26:06
19	-10.941532508608	-37.0573457786884	1	2014-09-13 10:26:12
20	-10.941532508608	-37.0573457786884	1	2014-09-13 10:26:18
21	-10.941532508608	-37.0573457786884	1	2014-09-13 10:26:23
22	-10.941532508608	-37.0573457786884	1	2014-09-13 10:26:28
23	-10.941532508608	-37.0573457786884	1	2014-09-13 10:26:33
24	-10.941532508608	-37.0573457786884	1	2014-09-13 10:26:38
25	-10.941532508608	-37.0573457786884	1	2014-09-13 10:26:44

Figure 28: A route instance and corresponding data.

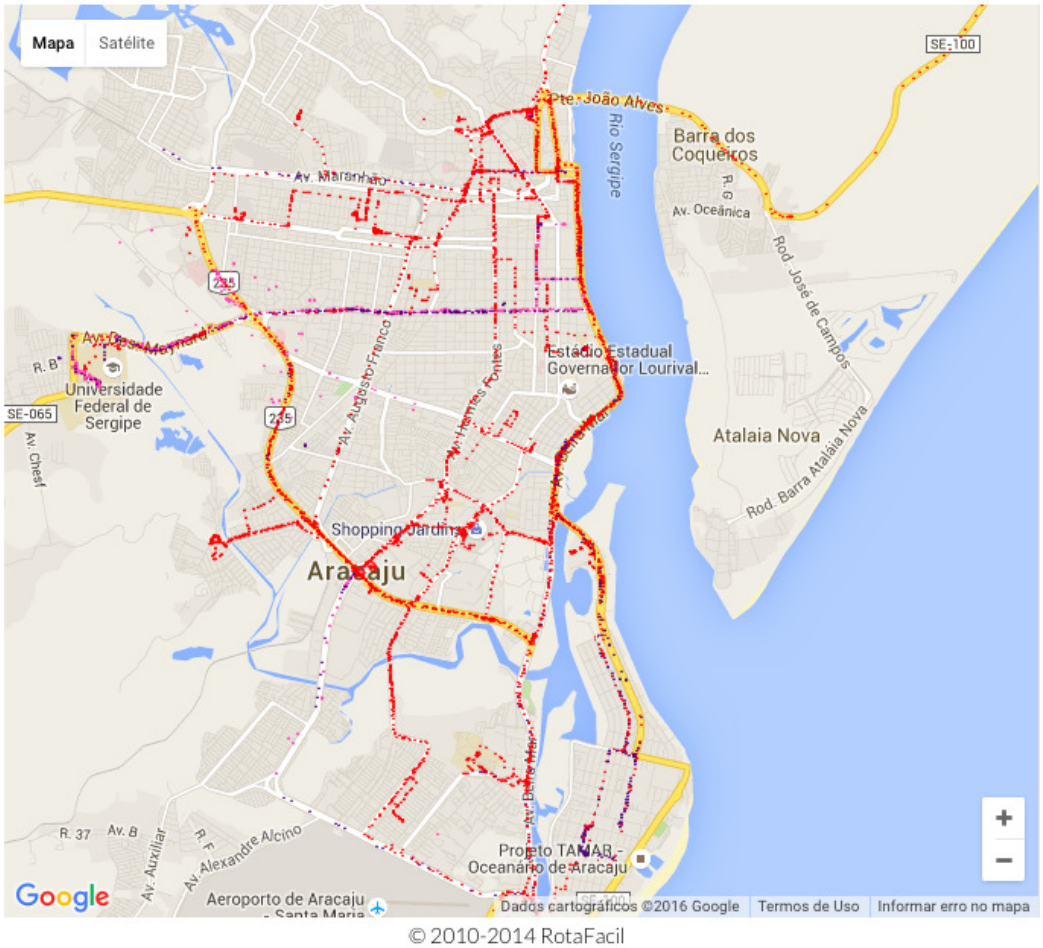


Figure 29: Streets and avenues of Aracaju city in the GO! Track dataset.

Table 12: Most visited traffic roads in Aracaju city according to GO!Track users.

Order	Address	Number Of Routes	Number Of Points
1	R. Boa Viagem	48	527
2	**Av. Pres. Tancredo**	38	1068
3	**Av. Beira Mar**	34	1216
4	Av. Ivo do Prado	23	581
5	Av. Mário Jorge Menezes Vieira	22	571
6	Av. Simeão Sobral	21	141
7	**Av. Des. Maynard**	21	950
8	**BR-235**	19	56
9	Av. Confiança	18	137
10	SE-100	18	280
11	Av. Eng. Gentil Tavares	14	253
12	**Av. Adélia Franco**	14	271
13	Av. Filadelfo Dória	14	34
14	Av. Dr. José da Silva Ribeiro Filho	14	184
15	R. de Muribeca	14	117
16	**Av. Barão de Maruim**	13	196
17	Av. Antônio Cabral	13	123
18	**Av. Coelho e Campos**	13	69
19	Av. Farmacêutica Cezartina Regis	13	158
20	Av. João Ribeiro	13	265

graph represents a route: the higher the line, more time was spent in the route regardless of the travelled distance. Points represent the city in a diverse range of times, allowing to note situations where the traffic is probably increased (peak time).

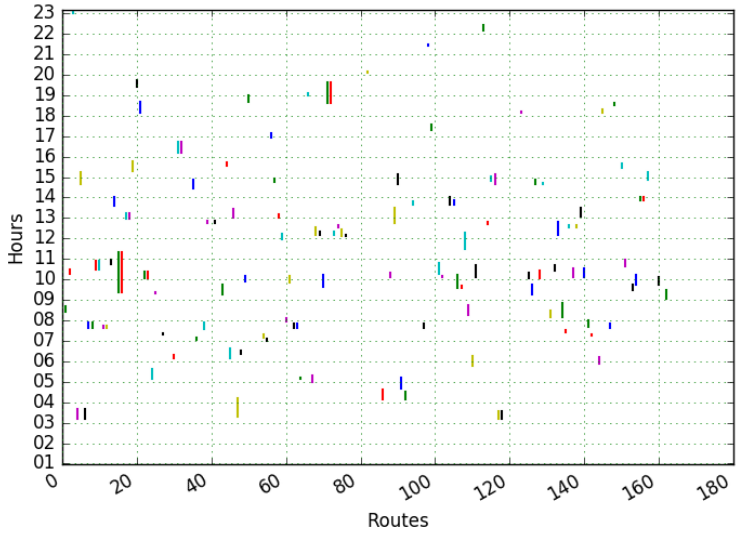


Figure 30: Relationship between routes and the date-time.

Similarly, the graph of Figure 31 shows the relation between each of the top-20

most visited traffic roads (street and avenues) and the specific instant it was visited. We have considered a daily time interval, in particular, at peak hours.

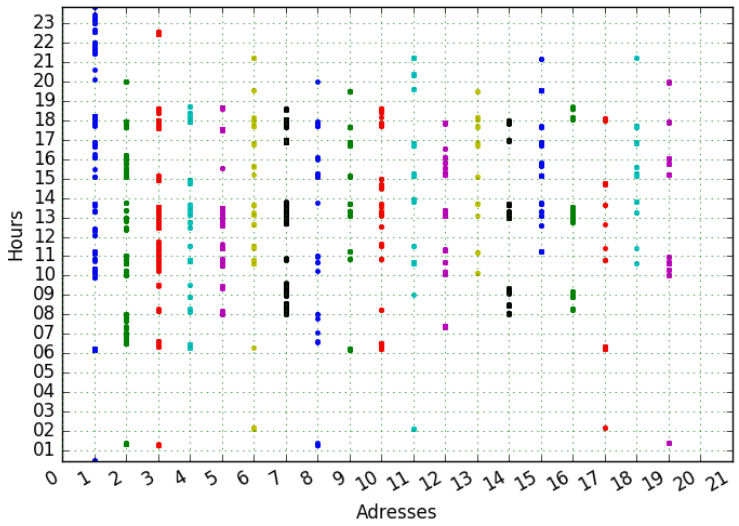


Figure 31: Relationship between traffic roads (streets and avenues) and the date-time.

4.3 Dataset Analysis

Firstly, we have analysed the dataset in regards to some basic statistics (table 13). A dataset histogram is also presented in Figure 32, which considers the amount of trajectories (y axis) and the length of the trajectories (x axis). The dataset present a right skewed distribution.

Table 13: Some basic statistics for GO! Track dataset.

Measures	values
Short trajectory	20.098
Bigger trajectory	57534.690
Standard deviation	11809.688
Mean	8276.297
Empty values	0

We have also tested the dataset with respect to its suitability for machine learning tasks. We have provided two clustering examples and a classification example. Optics algorithm (ANKERST et al., 1999) and K-Means (MACQUEEN et al., 1967) have been used to find clusters of trajectories that have any degree of similarity considering spatial

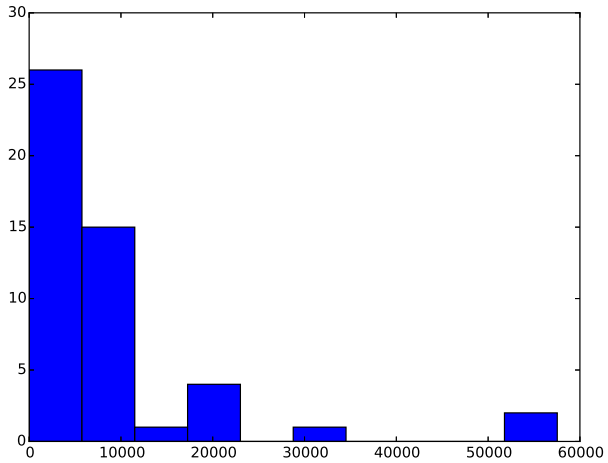


Figure 32: Dataset histogram.

distance. Furthermore, classification algorithm K-NN (K-Nearest Neighbors) has been used to classify such an example into the most suitable type of vehicle (bus or car).

4.3.1 Clustering task

Our approach to Optics and K-means works as follows.

Given a set of trajectories $C = \{Tr_1, Tr_2, ..., Tr_n\}$, it generates a set of clustering $A = \{C_1, C_2, ..., C_n\}$, where each C_i has similar trajectories. Trajectory is defined as $Tr_i = p_1, p_2, ..., p_n$, where each p_i is a triple (latitude, longitude,time). In addition, we consider some steps before clustering (CRUZ; MACEDO; GUIMARãES, 2015): (i) trajectory discretization and (ii) temporal filter.

The trajectory discretization has been used to save processing time, since the dataset C has a huge number of points collected in a short time interval. There are points with redundant information. Trajectory's discretization consists in computing a subset of points that is representative enough of the trajectory. Temporal filter was established so that only the set of trajectories with similar departure and destination times are eligible for the processing pipeline in order to avoid processing waste.

Optics is a density algorithm defined by three parameters considering cluster extraction: (i) ε, the radius of the search, (ii) *MinPts*, which is the minimum number of neighbours that defines a cluster and (iii) $\varepsilon^{'}$, that defines a distance bound to get clustering, $\varepsilon^{'} \leq \varepsilon$. The number k of the K-means has been defined randomly.

Table 14 shows some clustering results.

Figure 33 illustrates two trajectories that have been though to be similar by the clustering model.

Davies-Boundin index (DBI)(THEODORIDIS et al., 2010) has been used to evaluate the clustering task. It verifies how similar the elements of the same cluster are as well as the dissimilarity among distinct clusters.

Table 14: Results for the dataset with actual trajectories.

*Optics**	ε	*MinPts*	ε'	NC	*DBI*
	200	2	150	3	0.2885
	200	3	150	0	0
	300	2	250	3	0.2885
	300	3	250	0	0
K-means				k	*DBI*
				3	0.03571
				5	0.03529
				4	0.03499
				2	0.03665

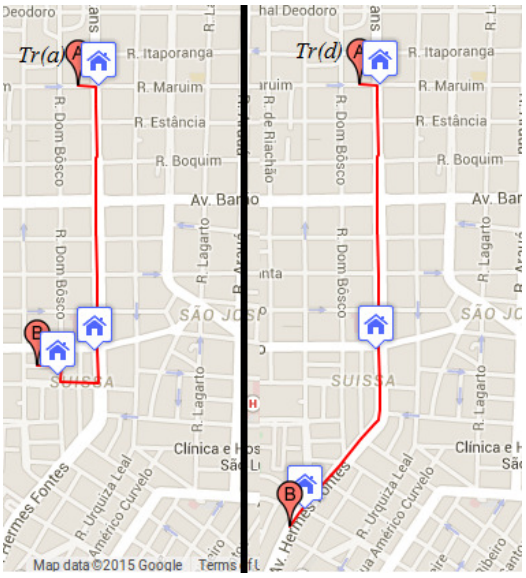


Figure 33: Similar trajectories that belong the same cluster.

4.3.2 Classification task

K-NN algorithm has been used to classify the type of vehicle according to the *speed* and *time* of travel attributes.

The data set has been divided into two subsets: (i) training set (80% of data) and (ii) test set (20% of data). We have empirically set $k = 15$. Figure 34 shows the scatter plot of the data.

According to the experiment, it was possible to classify the vehicle into a bus or a car. The error rate was 5%. Figure 35 shows a plot of the model generated by the algorithm.

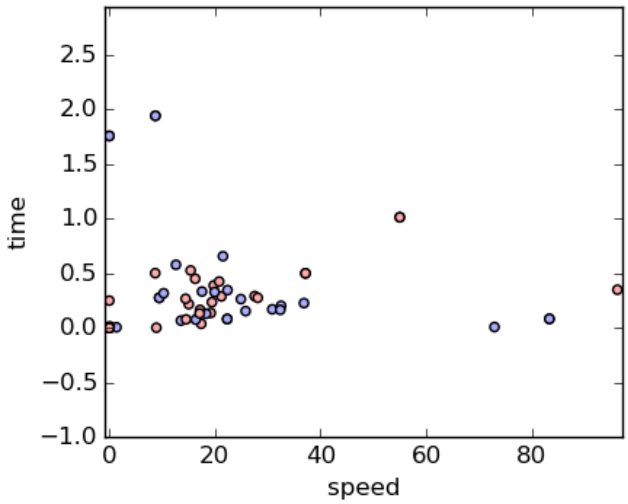


Figure 34: Scatter plot of speed and time

The red color represents the car class and the blue color represents the bus. According to confusion matrix, false negative rate was 5.8% and false positive rate was 4.7

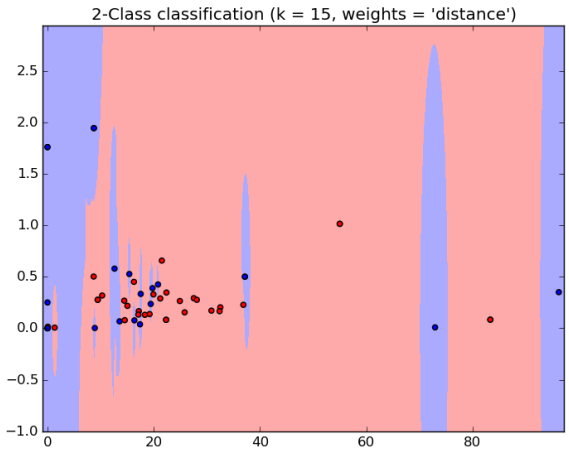


Figure 35: Classification of the type of vehicle.

4.4 Applications

Data obtained through GPS like shown in this work may be used with a plenty of goals. GPS data can be used to discover average speed of the roads in determined hour of the day. Average speed can be discovered through the historical data of each road that is enabled in dataset. Another information like estimate car flow on the road can be obtained through the historical data too, and, besides, it is possible to use some kind of regression to estimate future car flows or model a neural network to learning the dynamic of car flow etc. The quantity of information or applications that can be discover or develop with the

dataset is very large and this is essential to improve the quality of urban mobility and consequently to turn cities more smart.

Another information or applications can be thought like to verify which part of the day there is more car flow, estimating arrive time of the bus in specific place, verifying correlation between weather and bus speed, estimate time travel and recommend the best route in specific time of the day are more examples that the dataset with data about road traffic can help to understand and planning better strategies improve urban mobility.

4.5 Conclusion

Smart cities has been pointed as way to resolve some legacy problems related to urban mobility. Turning a city into a smart one requires a great effort, which involve from automatic urban traffic management until providing tools that enable citizens to monitoring and contributing to city planning. Even though urban mobility is a hot topic in research, open real traffic data are scarce.

In our work, we have presented an urban mobility dataset and the application used to build the dataset. We described the dataset structure and presented functionalities of the application. We used some basic static metrics to evaluate the dataset and applied some machine learning algorithms. From the analysis of results, we concluded that the dataset is useful and others information can be discovered.

As future works, we need to collect more data to generate a new version of this dataset. A second version of this dataset is already in production with the objective to increase the number of routes by range hours and the number of the individual users of the system. Another promising direction is to implement some applications present in section 6 to show the viability of the new dataset.

5 Conclusion

This dissertation had as main goals proposing and evaluating a new approach to cluster users in the context of ridesharing, that approach taking into account users trajectories, social and demographic profiles. Such approach was incorporated in a social network which permits users to register and publish their daily trajectories in order to find potential candidates to a ride and enables users to search for another which is interested in giving a ride. Contributing to a mechanism to help cities to improve the urban mobility and, consequently, soften the negative effects of traffic congestion in the emotional and physical health of citizens was the motivation for this work.

The problematic in question have some other problems to consider. They were noted and treated during the research time. By this reason, it was decided to organize the dissertation in self-contained chapters, without concerns with any type of temporal dependencies. Hereafter, we summarize the main aspects and contributions of each chapter.

In chapter 2, "GO!Caronas: fostering ridesharing with on-line social network, candidates clustering and ride matching", we presented an extended social network called GO!Caronas. We remodeled the social network GO! and added the functionality of ridematching proposed in chapter 3. We also presented some motivations to implement such application, described the architecture of the system and implemented three important concepts, such as ITS, ICT and carpooling.

The chapter 3, "Measuring the relevance of trajectory matching and profile matching in the context of carpooling computational systems" extended the approach proposed in (CRUZ; MACEDO; GUIMARãES, 2015) in order to consider user's profile as another feature to generate clusters. The new approach also formalized fundamental elements which make part of ridesharing. The elevated complexity of runtime presented by the new approach is a limitation and other important restriction was to consider the same weights for trajectory and profile.

The last chapter, "An open urban mobility datatset", presented an open real traffic dataset. The main contribution of this chapter is to make available a structured dataset. The chapter also presented a naive analyze of the dataset in order to verify the quality of the data and we showed some applications which can be gotten through the data. The application called GO!Track which is used to feed the dataset is also described.

The evaluation of the quality of user's clusters is a limitation considering the small amount of data which belongs to Go!Track dataset. Actions of encouraging to use GO!Track by part of the community has been done. Among them, the incorporation of useful functionalities to the population, such as related to bus transportation. This work

has been done. A deepest investigation about others ways to combine trajectory and profile clusters should be performed. Adaptation to use some techniques like boosting and bagging is an important direction.

This dissertation has some important limitations and, certainly, should be worked in a future investigation: an adequate evaluation about the usage of GO!Coronas application through the great part of the community in order to judge the feasibility of the approach related to interests and disposition of population in considering the ride sharing culture as a real alternative way to commute daily. This evaluation will enable to answer important questions, elaborate and confirm some hypothesis in order to conclude some correlations: Does the group concept encourage people to use ridesharing applications? Which are the parts of the cities more sensible to ridesharing practices? Which are the further solicited trajectories? Is there a relation between urban and intercities trips? Which is the group social more available to use ridesharing applications? Which are the season and period of the day more active? Finally, after some time of collecting data, it could be possible to conclude about the real impact of the application usage in urban mobility in a specific city. The verification method could consider an online questionnaire with a significant participation of the population about any improvements, observation about congestion indices in selected points of the city through the well-established applications like Waze or GoogleMaps, requisition of the officials data of the transportation department.

Bibliography

AGATZ, N. et al. Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research*, Elsevier B.V., v. 223, n. 2, p. 295–303, dez. 2012. ISSN 03772217. Disponível em: <http://linkinghub.elsevier.com/retrieve/pii/S0377221712003864>. Citado 3 vezes nas páginas 25, 27, and 55.

ALLAMARAJU, S. *RESTful Web Services Cookbook*. [S.l.]: O'Reilly Media, 2010. ISBN 9781449388843. Citado na página 34.

ANKERST, M. et al. Optics: ordering points to identify the clustering structure. In: ACM. *ACM Sigmod Record*. [S.l.], 1999. v. 28, n. 2, p. 49–60. Citado 4 vezes nas páginas 37, 48, 54, and 64.

AZZAM, E. R.; BELLIS, F. D. D. *Carona Brasil*. 2008. Http://www.caronabrasil.com.br/. [Accessed: 06-September-2013]. Citado 2 vezes nas páginas 27 and 32.

BARDIN, N. et al. *RideWith*. 2015. Https://play.google.com/store/apps/details?id=com.ridewith. [Accessed: 30-July-2015]. Citado na página 27.

CARVALHO, L. A.; MACEDO, H. T. Generation of coalition structures to provide proper groups' formation in group recommender systems. In: INTERNATIONAL WORLD WIDE WEB CONFERENCES STEERING COMMITTEE. *Proceedings of the 22nd international conference on World Wide Web companion*. [S.l.], 2013. p. 945–950. Citado na página 56.

CET. *CET - Rodízio Municipal*. 2013. Http://www.cetsp.com.br/consultas/rodizio-municipal/como-funciona.aspx. [Accessed: 14-September-2013]. Citado na página 26.

CHAN, N. D.; SHAHEEN, S. A. Ridesharing in north america: Past, present, and future. *Transport Reviews*, Taylor & Francis, v. 32, n. 1, p. 93–112, 2012. Citado 2 vezes nas páginas 31 and 32.

CHANG, Y. et al. Unsupervised video shot detection using clustering ensemble with a color global scale-invariant feature transform descriptor. *Journal on Image and Video Processing*, Hindawi Publishing Corp., v. 2008, p. 9, 2008. Citado na página 45.

CINTRA, M. Os custos dos congestionamentos na cidade de são paulo. 2014. Citado na página 25.

CRUZ, M. O.; MACEDO, H.; GUIMARãES, A. P. Grouping similar trajectories for carpooling purposes. In: *Brazilian Conference on Intelligent Systems*. [S.l.: s.n.], 2015. p. 234–239. ISBN 9781509000166. Citado 13 vezes nas páginas 29, 31, 36, 37, 43, 45, 46, 48, 50, 52, 55, 65, and 69.

DENATRAN. *DENATRAN - Departamento Nacional de Trânsito. Frota de veículos*. 2013. <http://www.denatran.gov.br/frota2013.htm>. [Accessed: 10-January-2014]. Citado na página 25.

DOYTSHER, Y.; GALON, B.; KANZA, Y. Storing routes in socio-spatial networks and supporting social-based route recommendation. In: ACM. *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks*. [S.l.], 2011. p. 49–56. Citado na página 44.

ERIKSSON, H.-E.; PENKER, M. Business modeling with uml. *Business Patterns at Work, John Wiley & Sons, New York, USA*, 2000. Citado na página 35.

FABER, Y. *Zaznu*. 2014. Http://www.zaznu.co/. [Accessed: 12-December-2015]. Citado na página 27.

FRED, A. L. N.; JAIN, A. K. Data clustering using evidence accumulation. *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, v. 4, p. 276–280, 2002. ISSN 10514651. Disponível em: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1047450>. Citado na página 45.

FROST, S. *Carticipate*. 2015. Http://www.carticipate.com/. [Accessed: 30-July-2015]. Citado na página 31.

FUNK, J. L. It and sustainability: new strategies for reducing carbon emissions and resource usage in transportation. *Telecommunications Policy*, Elsevier, v. 39, n. 10, p. 861–874, 2015. Citado na página 25.

FURUHATA, M. et al. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, Elsevier Ltd, v. 57, p. 28–46, nov 2013. ISSN 01912615. Disponível em: <http://linkinghub.elsevier.com/retrieve/pii/S0191261513001483>. Citado 3 vezes nas páginas 27, 44, and 55.

G1, S. P. *São Paulo tem congestionamento recorde, com 300 km de filas*. 2013. Http://glo.bo/1bWMGE7. [Accessed: 14-September-2013]. Citado na página 26.

GHAEMI, R. et al. A Survey : Clustering Ensembles Techniques. *Engineering and Technology*, v. 38, n. February, p. 636–645, 2009. ISSN 2070-3740. Disponível em: <http://www.akademik.unsri.ac.id/download/journal/files/waset/v50-109.pdf>. Citado na página 45.

GHOSEIRI, K. et al. *Real-time rideshare matching problem*. [s.n.], 2011. Disponível em: <http://ntl.bts.gov/lib/44000/44900/44921/UMD-2009-05.pdf>. Citado 2 vezes nas páginas 26 and 27.

GOOGLE. *Google Play Store*. 2013. <https://play.google.com/store/apps/>. [Accessed: 20-May-2015]. Citado na página 51.

GOWRI, R. *Car Pooling and Car Sharing: Simple Solution to Solve Complex Issues*. 2008. Http://www.frost.com/prod/servlet/market-insight-print.pag?docid=145008006. [Accessed: 14-September-2013]. Citado na página 26.

HE, W.; HWANG, K.; LI, D. Intelligent Carpool Routing for Urban Ridesharing by Mining GPS Trajectories. v. 15, n. 5, p. 2286–2296, 2014. Disponível em: <http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=6812>. Citado 4 vezes nas páginas 25, 26, 29, and 44.

HE, W.; HWANG, K.; LI, D. Intelligent carpool routing for urban ridesharing by mining gps trajectories. *Intelligent Transportation Systems, IEEE Transactions on*, IEEE, v. 15, n. 5, p. 2286–2296, 2014. Citado 3 vezes nas páginas 26, 27, and 28.

HE, W.; HWANG, K.; LI, D. Intelligent Carpool Routing for Urban Ridesharing by Mining GPS Trajectories. v. 15, n. 5, p. 2286–2296, 2014. Disponível em: <http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=6812>. Citado na página 27.

Michael Oliveira da Cruz. Maria Luiza Souza Matos Adolfo Guimaraes Hendrik Texeira Macedo. *GO!* 2014. BR 51 2014 000963 7. Citado na página 56.

HERRERA, J. C. et al. Evaluation of traffic data obtained via gps-enabled mobile phones: The mobile century field experiment. *Transportation Research Part C: Emerging Technologies*, Elsevier, v. 18, n. 4, p. 568–583, 2010. Citado na página 57.

HONG, Y. et al. Unsupervised feature selection using clustering ensembles and population based incremental learning algorithm. *Pattern Recognition*, v. 41, n. 9, p. 2742–2756, 2008. ISSN 00313203. Citado na página 45.

IQBAL, A.; MOH'D, a.; KHAN, Z. Semi-supervised clustering ensemble by voting. *arXiv preprint arXiv:1208.4138*, p. 1–5, 2012. Disponível em: <http://arxiv.org/abs/1208.4138>. Citado na página 45.

KALANICK, T.; CAMP, G. *Uber*. 2015. Https://www.uber.com/. [Accessed: 30-July-2015]. Citado 2 vezes nas páginas 27 and 31.

KELLEY, K. Casual carpooling—enhanced. *Journal of Public Transportation*, v. 10, n. 4, p. 119–130, 2007. Disponível em: <http://www.worldtransitresearch.info/research/3472/>. Citado na página 28.

KUHN, H. W. The hungarian method for the assignment problem. *Naval research logistics quarterly*, Wiley Online Library, v. 2, n. 1-2, p. 83–97, 1955. Citado na página 49.

LEE, J.-g.; HAN, J. Trajectory Clustering : A Partition-and-Group Framework. 2007. Citado 3 vezes nas páginas 27, 28, and 29.

LERNER, W.; AUDENHOVE, F.-J. V. The future of urban mobility: Towards networked, multimodal cities in 2050. *Public Transport International-English Edition*, v. 61, n. 2, p. 14, 2012. Citado na página 25.

LEVIN, I. P. et al. Measurement of psychological factors and their role in travel behavior. *Transportation Research Record*, Transportation Research Board, National Research Council, National Academy of Sciences USA, v. 649, p. 1–7, 1977. Citado na página 27.

LEVY, J. I.; BUONOCORE, J. J.; STACKELBERG, K. von. Evaluation of the public health impacts of traffic congestion: a health risk assessment. *Environmental health : a global access science source*, BioMed Central Ltd, v. 9, n. 1, p. 65, jan 2010. ISSN 1476-069X. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2987789{&}tool=pmcentrez{&}rendertype=ab>. Citado na página 25.

MACQUEEN, J. et al. Some methods for classification and analysis of multivariate observations. In: OAKLAND, CA, USA. *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. [S.l.], 1967. v. 1, n. 14, p. 281–297. Citado 2 vezes nas páginas 43 and 64.

MATOS, M. L. et al. A social network for carpooling. In: ACM. *Proceedings of the 7th Euro American Conference on Telematics and Information Systems*. [S.l.], 2014. p. 10. Citado 6 vezes nas páginas 29, 31, 33, 34, 39, and 56.

MAZZELLA, F. *BlaBlaCar*. 2004. Http://www.blablacar.com. [Accessed: 06-September-2013]. Citado 2 vezes nas páginas 27 and 31.

MENG, F.; TONG, X.; WANG, Z. A clustering-ensemble approach based on voting. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, v. 7002 LNAI, n. PART 1, p. 421–427, 2011. ISSN 03029743. Citado 2 vezes nas páginas 43 and 45.

O'SULLIVAN, S. *Carma*. 2015. Https://carmacarpool.com. [Accessed: 30-July-2015]. Citado na página 31.

SCHRANK, D.; EISELE, B.; LOMAX, T. *2014 Urban Mobility Report: Powered by INRIX Traffic Data*. [S.l.], 2015. Citado na página 25.

STREHL, A.; GHOSH, J. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, JMLR. org, v. 3, p. 583–617, 2003. Citado na página 45.

TELES, R. et al. Automatic generation of human-like route descriptions: a corpus-driven approach. *Journal of Emerging Technologies in Web Intelligence*, v. 5, n. 4, p. 413–423, 2013. Citado na página 46.

TELLES, R.; GUIMARÃES, A. P.; MACEDO, H. T. Automated Feeding of POI Base for The Generation of Route Descriptions. In: *Euro-American Conference on Telematics and Information Systems, EATIS 2012*. [S.l.: s.n.], 2012. p. 253–259. Citado na página 46.

THEODORIDIS, S. et al. *Introduction to Pattern Recognition: A Matlab Approach: A Matlab Approach*. [S.l.]: Academic Press, 2010. Citado 3 vezes nas páginas 49, 51, and 66.

VAXMAN, A. et al. *tripda.com.br*. 2014. Http://www.tripda.com.br/. [Accessed: 12-November-2014]. Citado 2 vezes nas páginas 27 and 31.

YAN, S.; CHEN, C.-Y. A model and a solution algorithm for the car pooling problem with pre-matching information. *Computers & Industrial Engineering*, Elsevier, v. 61, n. 3, p. 512–524, 2011. Citado 2 vezes nas páginas 43 and 44.

YAN, Z. et al. Semitri: a framework for semantic annotation of heterogeneous trajectories. In: ACM. *Proceedings of the 14th international conference on extending database technology*. [S.l.], 2011. p. 259–270. Citado na página 57.

YANAGI, Y.; ASSUNÇÃO, J. V. d.; BARROZO, L. V. The impact of atmospheric particulate matter on cancer incidence and mortality in the city of sao paulo, brazil. *Cadernos de Saúde Pública*, SciELO Public Health, v. 28, n. 9, p. 1737–1748, 2012. Citado na página 25.

ZHENG, Y.; XIE, X.; MA, W. GeoLife: A Collaborative Social Networking Service among User, Location and Trajectory. *IEEE Data Eng. Bulletin*, v. 33, n. 2, p. 32–40, 2010. Disponível em: <http://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:GeoLife+:+A+Collaborative+Social+Networking+Service+among+User+,+Location+and+Trajec>. Citado 2 vezes nas páginas 28 and 57.

ZIMMER, L. G. . J. *Lyft*. 2015. Https://www.lyft.com/. [Accessed: 30-July-2015]. Citado 2 vezes nas páginas 27 and 31.

APPENDIX A – UCI Machine Learning Repository

Donation Form:

1. Dataset Name: GPS Trajectories;
2. Data Type: Multivariate;
3. Task: Classification, Regression, Clustering;
4. Attribute Type: Categorical, Real;
5. Area: CS or Engineering;
6. Format Type: Non-Matrix;
7. Missing Values: False;
8. Number of Instances: 163;
9. Number of Instances: 163;
10. Number of Attributes: 15;
11. Attributes Information:
    - id: unique key to identify each trajectory;
    - id-android: an identifier for each device that was used to collect trajectories. This attribute does not have any relation to real device;
    - speed: average speed during all pathway;
    - time: the duration of the pathway in minutes;
    - distance: distance of the trajectory in kilometer;
    - rating: the user evaluation about the traffic;
    - rating-bus: indicates the quality of the travel (available just in bus case);
    - rating-weather: indicates the conditions of the weather (available just in bus case);
    - car-or-bus: indicates if the route was collected by a car or a bus;
    - line: information about the bus that does the pathway (available just in bus case).

The dataset is available on UCI [1]

[1] http://archive.ics.uci.edu/ml/datasets/GPS+Trajectories

APPENDIX B – GO!Track Manual

Este manual fornece as informações necessárias para uso do aplicativo GO!Track. O manual estará explicando as funcionalidades de cada tela que o aplicativo disponibiliza para o usuário.

B.1 Home Page

A home page possui duas opções de meio de transporte para o rastreamento de dados: Carro e Ônibus.

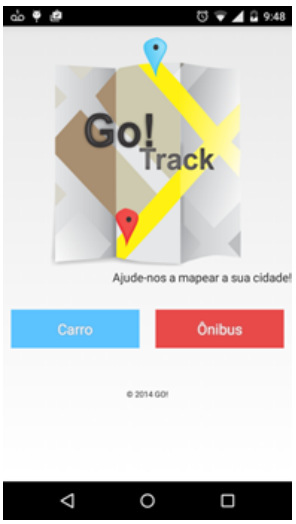


Figure 36: Escolha entre carro e ônibus

O usuário deve ativar o dispositivo GPS do celular para que o app possa funcionar corretamente.

B.2 Carro

A Figura 37 mostra a tela de rastreamento de dados por meio de veículo/carro. Neste tipo de rastreamento o usuário basta apenas apertar o botão start conforme a Figura 37.

Após apertar o botão stop o app solicitará que o usuário avalie a condição do trânsito de acordo com sua percepção e valorando-a com uma das seguintes opções: boa, regular e ruim.

Figure 37: Última localização do usuário

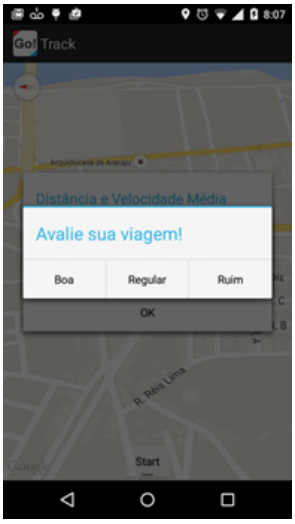


Figure 38: Avaliação da viagem

B.3 Ônibus

A Figura 39 mostra a tela de rastreamento de dados por meio de Ônibus. Neste tipo de rastreamento, o usuário pode escolher a linha de ônibus, que irá utilizar, e visualizar todos os pontos de ônibus da cidade de Aracaju conforme mostram as Figuras 39 e 40.

O aplicativo inicialmente mostra também a última localização do usuário por meio do último dado de GPS.

Após o acionamento do botão start, o aplicativo posiciona o usuário de acordo com sua nova localização e começa a capturar dados do trânsito conforme mostra a Figura 41.

A Figura 42 mostra que após apertar o botão stop, o aplicativo solicitará que o

Figure 39: Linhas de ônibus

Figure 40: Última localização

usuário faça uma avaliação sobre as condições do trânsito, tempo e lotação do ônibus.

B.4 Detalhes do Servidor

Atualmente, o *back-end* do aplicativo GO!Track está hospedado no *OpenShift*, que é uma plataforma pertencente a *Red Hat*. A Figura 43 mostra o diagrama de implantação do aplicativo GO!Track. O diagrama mostra detalhes de comunicação do aplicativo com servidor além do padrão arquitetural MVC (Model-View-Controller) utilizado para desenvolver o aplicativo.

Os dados são enviados para o servidor via JSON de acordo com o exemplo da Figura 44

Figure 41: Início do rastramento.

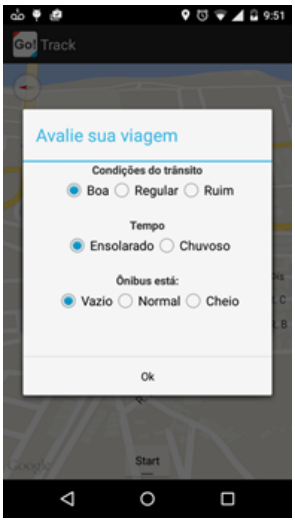


Figure 42: Avaliação da viagem.

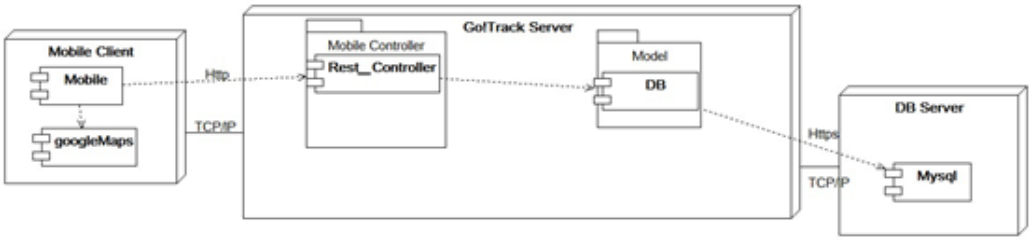


Figure 43: Diagrama de implantação GO!Track

B.5 Banco de Dados

O banco de dados utilizado no servidor é Mysql 4.5.1. A aplicação conta apenas com duas tabelas conforme ilustra a Figura 45.

```
{
"time": "2014-08-28T10:03:34Z",
"distance": 1.0,
"speed": 2.0,
"rating": 1,
"car_or_bus": 0,
"tracks_points": [
{"latitude": 1.0, "longitude": 2.0, "time": "2014-08-28T10:03:56Z"},
{"latitude": 2.0, "longitude": 1.0, "time": "2014-08-28T10:04:09Z"}
]
```


Figure 44: Exemplo json

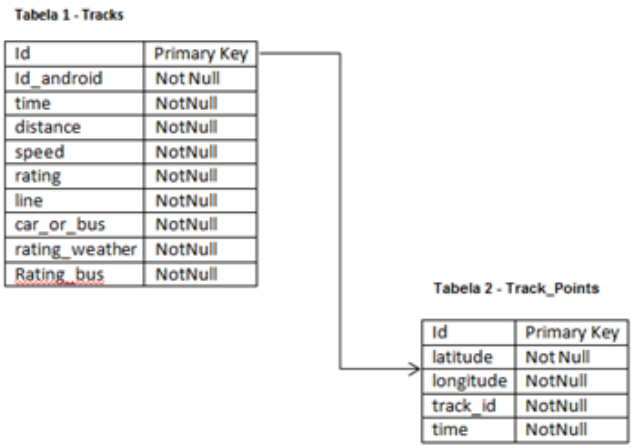


Figure 45: Diagrama ER GO!Track