Passenger density and flow analysis and city zones and bus stops classification for public bus service management

Raul S. Barth, Renata Galante

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS) – 91.501-970 – Porto Alegre – RS – Brazil

{rsbarth@inf.ufrgs.br, galante@inf.ufrgs.br}

Abstract. *This work presents, for the first time in literature, a low-cost framework to mine data obtained from passengers smart cards, buses GPS and bus stops geolocation using Lambda Architecture approach. Operators, companies, government and passengers will use this knowledge for improving usability, comfort, and quality of transportation service. This analysis gives greater insight into the volume and flow of passengers and the real existing demand for bus services, facilitating its control and management, allowing decision-making. As result, bus stops and city areas are classified according to buses demand.*

1. Introduction

The scope of this work falls within Smart Cities, a Big Data Analytics sub-area, that makes cities infrastructure and public services more iterative, efficient and intelligent [Pellicer, S. et al. 2013]. It belongs to a Smart Cities topic called Smart Mobility, focusing on analysis of flow and density of bus service passengers. [United Nations 2011] affirms that the world's urban population will increase from 2.6 billion in 2010 to 5.2 billion in 2050, about 70% of world population. [IPEA 2015] shows that 65% of Brazil population use public transportation. Population growth will bring many challenges to the cities, such as an increase in the number of public transport users. Improving quality of public bus service become essential. Therefore, passenger flow and density analysis provides great understanding of bus service network, showing its real demand and usage, allowing better control, management and decision-making.

[Qing Z. et al. 2009] uses data from Beijing Smart Card to show that it is an important source of information of passengers' behaviour. The passenger flow analysis, combining GPS data and Smart Card is proposed by [Duan W. et al. 2012]. [J. Zhang et al. 2014] uses the same combination of data sources, however, aims to calculate the passenger density of a bus service, showing a model of buses schedule table redefinition. In the context of flow analysis and passenger density there is no work focused on using GPS data and smart card together with city areas clustering. This work implements Lambda Architecture [Kiran, Mariam et al 2015] as the processing unit for computing passenger and bus data. The goal is to develop methods capable of analysing passenger flow and density, using data obtained from GPS, Smart card and bus stops geolocation along with clustering techniques. Therefore, characterize city areas and bus stops that will demonstrate a real scenario of public bus demand, enabling better governance and reorganization of the service, supplying users' needs.

The paper is organized as follows: Section 2 presents related works, containing the state of the art of this scenario and important papers; Section 3 presents the proposed method; Section 4 presents the case of study and its evaluation and Section 5 and 6 give the conclusion and the references, respectively.

2. Related Work

The state-of-the-art of urban mobility area of Smart Cities was analysed, mainly solutions and models for density and passenger flow analysis and Origin-Destination (OD) matrix computation. There are three main methods used to solve this problem: intersection of GPS data and Smart Card; separation of the bus lines in different segments to determine user's shipping segment; and usage of buses timetables. The first uses time-matching algorithms, but may become impractical to require that buses have GPS and Fare Collection Systems. The second has low precision results, while the third can be affected by several factors such as bus speed and traffic.

[Qing Z. et al., 2009] addresses the analysis of Smart Card without considering other data sources, to prove that it is a major source of information passenger traffic. [Li, M. et al. 2012] addresses the use of the second and third methods by proposing two steps: separation date card and bus stops matching. It uses the idea of Automatic Data Collection Systems, and aims to describe a method to determine the original location by associating bus card data and spatial relationship of stop lines or destination. [Duan W. et al., 2012] proposes to integrate data originated from GPS and Smart Card, with the goal of analysing passenger flow of single-line type for Beijing. [Zhang J. et al. 2014] presents for the first time the joint use of data from GPS and Smart Card to calculate passenger density on a bus service. It proposes a new method for fusion of GPS data and Smart Card that can reduce the timing error in the type Fare Collection Devices (FCD) data. [Kieu, Le Minh et al. 2015] uses Smart Card data to focus on passenger segmentation. For the first time in literature, the bus stops geolocation are used to aggregate information and to enable the knowledge of what bus stops the passenger exactly has boarded or gotten off. That permit a map-density of each stop from a bus line and, consequently, from all public bus service.

3. DMBSM – Data Mining Framework for Bus Service Management

Data Mining Framework for Bus Service Management (DMBSM) is a framework that has as input GPS, bus stop and smart card data and processes it extracting useful information as passengers' density and flow, and bus stops segmentation based on travel purposes. From results, the real bus service demand is known, enabling decision-making. DMBSM is based on Lambda Architecture, using Big Data tools such as Apache Hadoop and Apache Spark for parallel processing and performance increasing.

Problem Statement. A density-and-flow-map of a bus line will be built representing the real need and usage of buses, allowing decision-making for buses management and control. Results of the bus stops and city classification can be modelled, according to a defined precision radius representing the relation between the bus position and the bus stops location. It will be possible to define, for example,' that each bus stops owns a radius of 50 meters from its real location. It means that all GPS buses records that have a position in this radius precision belong to that bus stop. The model will also build a city characterization, aiming to categorize it in a scale of purposes areas types: residential, work/business, nightlife, and areas with small or high passengers' density and flow. These categories are pre-defined and chosen to be the most general as possible, representing the main city activities, without losing information. Based on that, government and bus service operators will have a real scenario of the public bus service usage, being able to make a better governance, management and reorganize buses and lines to meet passengers' needs, with comfort and quality. The travel proposes criteria

are pre-defined according to the daytime that the buses are taken. The direction of the bus must be considered - *incoming* or *outcoming* buses. For standardization proposes, we consider the intervals shown on *Table 1*. The time slots were built, for generalization, even we know some cities - especially metropoles – use to present discrepancy when compared to the slots – when having 24h industries and stores, or with an intense nightlife. *Income Time Interval* and *Outcome Time Interval* represent the time of buses arriving, or leaving, respectively, a city zone - or bus stop.

Table 1: Daytime interval for bus stop segmentation.

Segment Name	Income Time Interval	Outcome Time Interval
Residential	6pm to 8pm & 2am to 6am	6am to 8am
Work/Commercial	7am to 9am	5pm to 7pm
Night Life	8pm to 11pm	11pm to 5am

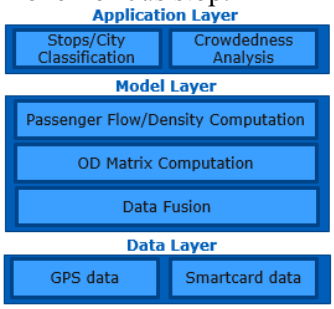

Figure 1: Approach steps.

Architecture. DMBSM reaches Lambda Architecture [Kiran, Mariam et al, 2015]. Three layers are involved in the model representation. Starting in a bottom-up approach, the first layer - *Data Layer* - contains the data extracted from the GPS and Fare Collection System - Smart Cards. Furthermore, it also contains the buses stops geolocation. Relating this layer with Lambda Architecture described above, it contains the message broker - Apache Kafka. The *Model Layer* represents the core of the project inasmuch as it is responsible for the massive computation. First, there is the *Data Fusion* module, which unifies the information from the *Data Layer*. Then the *Matrix OD Computation* module computes the Origen-Destination matrix of the passengers and with this information, the next module can calculate the passengers flow and density. This layer is the batch layer of Lambda Architecture, composed by Hadoop – Hadoop Distributed File System - Spark and MapReduce jobs. The goal is to allow parallel processing and performance increasing by multi-node clustering. After the processing layer, there is the *App Layer,* in a high level representing the analysis of the results and the crowdedness analysis. This last layer is represented by Apache Zeppelin - a web-based tool – to expose the analysis results in dashboards. Therefore, in general we have the data extraction - *Data Layer* - data mining and passenger's density and flow computation - *Model Layer* - and the result analysis and city segmentation - *App Layer.*

Data Analysis. Three different data types are used. GPS, containing as attributes *Bus ID, Time, Bus Lines, Latitude and Longitude*, buses stops geolocation, with attributes *Stop ID, Stop Code, Stop Name, Latitude, Longitude*, and Smart Cards, containing Fare Collection System data and composed by *Smartcard ID, Time, Transaction Type and Metro Station/Bus Line.* In GPS data, *Bus ID* represents the vehicle ID, *Time* is the record time, *Bus Lines* has the line of the recorded bus and *latitude* and *longitude* represent the position information of the vehicle. In the Smart Card data, *Smartcad ID* represents the ID of the tapped card, *Time* represents the moment the tapping information was recorded, *Transaction Type* - in our case - indicates that is a boarding, and *Bus Line* has the bus line - the Fare Collection System where the smart card data was recorded. Bus stop dataset can improve the accuracy and permit us to obtain better results once we know exactly the bus stops where the passenger boarded. *Bus Stop ID*

represents the station id, *Latitude* and *Longitude* represent the bus stop position and *Terminal* indicates if the recorded bus stops is a bus terminal with connections. Passenger origin derivation process is considerably easy since we consider that the time recorded in a tapping card stored Smart Card dataset represents the moment when the passenger has boarded. For that propose, attributes as *SmartCard Id, Time and Bus Line* from a card record are used. In comparison with literature [Zhang J. et al. 2014], this one also makes use of *time-matching method* to build the Origin-Destination (OD) passengers matrices. Once this computation is done, is possible to make inferences and to analysis the density and the flow, obtaining source of knowledge to segment the bus stops and city using clustering algorithms.

The first step is passengers' **density** and **flow** computation, aiming to build all Origin-Destination matrices. First, having the coordinates of each smart card record and bus stops, we can identify in what bus stop the passenger boarded. It is assumed the passenger tap the card as soon as he gets on the bus. Data pre-processing is necessary before starting the computation process. The GPS data and Smart Card data must be separated according to the target bus line to focus the computation process separately for all buses of each bus line. The next steps is the Smart Card ID and GPS ID pairs discovery, which represents the respective smart card record related to a GPS record. Thereunto, we have to compare the getting on station time and the tapping card time. Through this comparison and considering the minimum time difference between GPS record and Smart card record, we are able to find out the passenger getting-on station. GPS records that have their coordinates positioned inside a bus stop coverage radius belong to it. From that, we can pre-process and determine, for each bus position, if this bus is in a bus stops coverage radius. Then, the problem is limited to compare GPS and Smart Card times using *time-matching* algorithm. The process of obtaining the passenger destination requires some assumptions and it is an estimation process since we do not have smart card records for passengers getting off the buses. To allow this computation, time slots will be determined and classified according to purposes. Passenger's density represents the density – number – of passengers in inside the bus in each stop, which is incrementally computed by considering the number of passengers that boarded in the bus minus the ones that got off.

4. Case Study

This first experiment aims to demonstrate the feasibility of the proposed work. Moreover, passenger density and bus stops classification are performed to prove that this project can be further developed. Here, we also check the data availability and how the different datasets can be crossed to obtain the expected results. For each Smart Card record, its related bus stop and travel purpose were found. Bus stops were classified based on the main passengers travel purposes leaving the stop. Results give knowledge about the real bus service demand, showing passenger density and travel purposes.

Dataset. Three different data types are used: GPS (*Bus ID, Time, Bus Lines, Latitude and Longitude*) with 19485 records, buses stops geolocation (*Stop ID, Stop Code, Stop Name, Latitude, Longitude*) and Smart Cards (*Smartcard ID, Time, Transaction Type and Metro Station/Bus Line)* with 3186 records. Bus GPS and Smart Card datasets were obtained from [Schenzhen City in China], between 22, October 2013 - 6am - and 23, October 2013 - 12am, containing 400Mb.

Methodology. The first part consists in pre-process data. Here, we approach a specific line (B606) and datasets were filtered based on that. Then, all GPS records were linked to a bus stop, based on a time-matching algorithm. The algorithm considers the coordinates and the timestamp to find GPS-Stops pairs. GPS-Cards pairs were found, also based on time-matching. At the end, cards registers and stops were classified into one of the four categories: *residential, work, nightlife, personal.* As a first experiment to show the project feasibility and objective, the used dataset contains data of one day of the city of Shenzhen, in China. Although the experiment is limited, important steps were developed and all smart cards had been linked to a stop and related to a travel purpose. The same was done with the stops that were also classified.

First, some pre-processing steps must be performed to clean and filter data. The next step is a classification problem that consists in classifying all GPS records in their respective stops. Each GPS record coordinates are compared to all bus stops coordinates. It was done using a Spark job running inside Hadoop, in Lambda Architecture. For the experiment, the day intervals was divided in: *residential (*6am-8am), *work* (5pm-8pm), *nightlife* (11pm-5am) and *personal. Personal* category encompasses all the remaining periods not classified in the others. The target bus stop is the one that presents the minimum distance difference. Then, we had the passenger density computation and bus stop main travel purpose discovery. GPS x Card pairs are obtained through a time-matching method, comparing time attribute of GPS and Smartcard records.

Evaluation. Results can be seen in the three graphics below. *Figure 2 (L)* demonstrates how the cards are distributed among the travel purposes. From total 3186 cards registers analysed, 1350 – 42.4% - represent *residential* as the main travel purpose. *Work* appears as the second main reason for taking a bus, with 1020 – 32% - departing buses, and *personal* appears with 816 – 25.6% - records. It is important to mention that the *personal* category was added in this first experiment as the datasets represent only one day in the bus service network. *Night* purpose did not have any record.

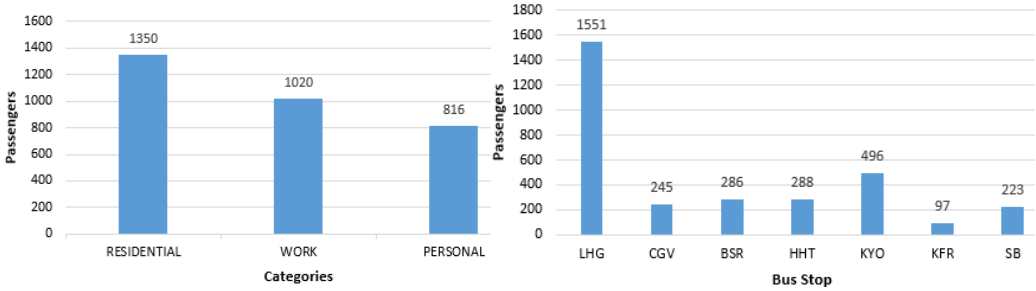

Figure 2: Smart card records classification (L) and Passenger density per bus stops (R)

Figure 2 (R) shows the passenger density in each bus stop. *LHD* is the densest stop, with 653 records, being 379 of *work* type and 274 of *residential* type. *KFR* appears as the less dense stop, with only 46 passengers leaving the stop, 22 of them with *work* as travel purpose and 24 with *residential* as travel purpose. *Figure 3 s*hows the density of the second graph distributed between three travel purposes: *work, residential* and *nightlife.* Again, we see *LHG* stop with the highest amount. From that graph, we can notice that this stop is the one that has more passengers leaving the station in the peak periods – 6am to 8am for *residential* and 5pm to 7pm for *work,* which result in the highest passenger density shown in the second graph.

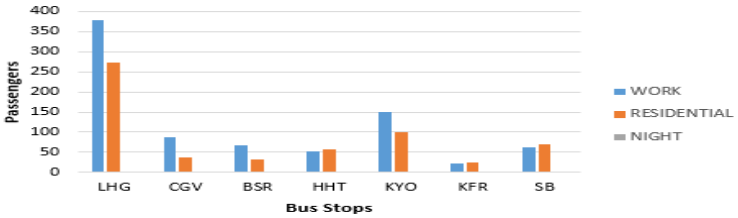

Figure 3: Bus stop classification

5. Conclusion

The improvement on public transport quality influences directly and positively in society. Furthermore, it brings countless benefits for people lives as well as helps to solve urban mobility problems that are current present in big cities. Knowledge about public transport network behaviour allows decision-making, increasing, service quality, passenger experience, usage and profits. In this paper, we described DMBSM, a data-mining framework that permits public service management, through passenger density and flow analysis, as well as bus stops and city zones segmentation. Using concepts of Big Data, Smart Cities and Lambda Box Architecture – Apache Kafka, Hadoop and Spark - the model computes passengers' density and flows using time-matching methods and clustering. As used datasets were data-limited, the experiment did not represent the whole scope of this project, but a part of it. However, it demonstrates the feasibility of what is proposed; showing that understanding passengers' behaviour, travel proposals, density and flow are an important source of knowledge to improve public bus service.

Future work will implement Lambda Architecture containing a batch layer to make parallel data processing and store it in Hadoop, as well as a speed layer running Spark Streaming for streaming processing. Days will be divided into business days and weekend days. Therefore, *NightLife* category can be more assertive in the sense that they will be considered just for Friday, Saturday and Sunday.

6. References

United Nations (2011). "Population Distribution, urbanization, internal migration and development: An international perspective"

IPEA (2015) - Instituto de Pesquisa Econômica Aplicada.

Pellicer, S. et al. (2013) "A Global Perspective of Smart Cities: A Survey".

Qing, Z. et al. (2009) "Public Transport IC Card Data Analysis and Operation Strategy Research Based on Data ining Technology"

Li, Man et al. (2012) "Public Transport Smart Card Data Analysis and Passenger Flow Distribution"

Duan, W. et al. (2012) "Analysis of Single-line Passenger Flow Based on IC Data and GPS Data"

Zhang, J. et al. (2014) "Analysing Passenger Density for Public Bus: Inference of Crowdedness and Evaluation of Scheduling Choices"

Kieu, Le Minh; Bhaskar, Ashish; Chung, Edward (2015) "Passenger Segmentation Using Smart card Data"

Kiran, Mariam et al. (2015) "Lambda architecture for cost-effective batch and speed big data processing"

Shenzhen, in China "Sample Data Description of mPat - http://cloud.siat.ac.cn/mpat/"