both entities (head and tail) for each triple predicted as correct in $\mathcal{T}_0$ with its k nearest neighbors. Then, we obtain $\hat{\mathcal{G}}_0^k = \{g(x_{h',r,t'}) \mid x_{h,r,t} \in \mathcal{T}_0^+ \wedge e_{h'} \in N_k(e_h \mid \Theta_g) \wedge e_{t'} \in N_k(e_t \mid \Theta_g)\}$ that is the set of correct triples inferred by the classifier $g(\cdot)$, where $N_k(e \mid \Theta_g)$ represents the subset constructed by e and its $k-1$ nearest neighbors, and Θ_g is the vector formed by the entities.

III. Experiments

In a previous work [2], Gusmão et. al applied **XKE-PRED** and **XKE-TRUE** over the results of the embedding model **TransE** learned over the datasets **FB13** and **NELL186** (shown Table I). In this paper, we expand their work by applying both XKE variants to **Analogy** models learned over the same datasets, aiming to investigate and compare performances. Both embedding models were trained via grid search, limited to 1.000 epochs, and negative triples were generated using the Bernoulli distribution method in a rate of two negative triples for each positive example. Even tough we tested entity similarity for nearest neighbors using $k = 3$, 5 and 7, due to space constraints we report only results for $k = 5$.

Regarding performance, metrics we consider are:

- fidelity: the ability of the pedagogical method to mimic the behavior of the embedding model. More formally, let us denote $\mathcal{D}_{test}$ a set of test triples. Let $\hat{y}_g$ denote a prediction made by an embedding model for a given input triple $\langle e_h, r_r, e_t\rangle \in \mathcal{D}_{test}$ and $\hat{y}_{XKE}$ the prediction made by XKE for the same triple. Fidelity can be computed as $\mathbb{P}(\hat{y}_{XKE} = \hat{y}_g) = \sum_{x\in\mathcal{D}_{test}} \frac{1-|\hat{y}_{XKE}-\hat{y}_g|}{|\mathcal{D}_{test}|}$.

- accuracy: the ability of the weighted rules to correctly predict real data. More formally, let $\hat{y}$ denote a prediction of a given classifier, y a binary random variable that models the existence of the same instance, and $\mathcal{D}_{test}$ a test set. Accuracy can be computed by $\mathbb{P}(\hat{y} = y) = \sum_{x\in\mathcal{D}_{test}} \frac{1-|\hat{y}-y|}{|\mathcal{D}_{test}|}$.

And for interpretability, two metrics are considered:

- # of rules with weight greater than zero;

- mean rule lenght.

The embedding models' performance is presented in TABLE II. With regards to accuracy, Analogy over FB13 had a much worse performance than TransE, whilst over NELL186, it performed much better. We suspect that the intrinsic topology of each dataset directly influences the embedding model performance, but this analysis is beyond of the scope of this paper. We can see that, for $\hat{\mathcal{G}}$ triple generation, Analogy performed better than TransE for XKE-PRED. This increase in performance was dramatic in FB13 and more subtle in NELL186. Regarding accuracy, we can see that, considering only examples with more than one feature extracted, both methods obtained a similar performance in FB13 and a better performance in NELL, similar behavior found for fidelity. With regard to interpretability, the explanation mean rule is less than four, due to a constraint imposed for performance reasons, and the number of rules extracted remained almost the same for FB13, and decreased for NELL186, turning the interpretation simpler using Analogy in contrast to TransE.

IV. Conclusion

We can see that both XKE-PRED and XKE-TRUE can be applied over datasets FB13 and NELL186 using Analogy embedding model with good results, meaning that XKE can model the new facts inferred by the embedding model with good fidelity. So far we tested only two datasets and two different embedding models, for future work, expanding this analysis would be a good line of research. Another interesting line would be to investigate the intrinsic structure of different benchmark datasets, to understand how its differences influence both the performance of the embedding model itself, as well as the performance of XKE.

Acknowledgement

This work was accomplished with the support from Itaú Unibanco S.A., through the scholarship program *PBI - Programa de Bolsas Itaú* together with the *C2D - Centro de Ciência de Dados* from *EPUSP - Escola Politécnica da Universidade de São Paulo*. Second author is supported by an agreement between IBM and the Sao Paulo Research Foundation (FAPESP) (grant 2017/19007-6). The project received financial support from FAPESP (grant 2016/18841-0). The third author is partially supported by CNPq (grant 308433/2014-9).

References

[1] WANG, Q. et al. Knowledge Graph Embedding : A Survey of Approaches and Applications. *IEEE Transactions on Knowledge and Data Engineering*, v. 29, n. 12, p. 2724–2743, 2017.

[2] GUSMÃO, A. C. et al. Interpreting Embedding Models of Knowledge Bases : A Pedagogical Approach. n. Whi, 2018.

[3] GUNNING, D. *Broad Agency Announcement Explainable Artificial Intelligence (XAI)*. [S.l.], 2016. DARPA–BAA–16–53 p.

[4] RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. 2016. ISSN 9781450321389.

[5] MITCHELL, T. et al. Never-Ending Learning. *Communications of the Acm*, v. 61, n. 1, p. 2302–2310, 2015.

[6] SUCHANEK, F. M.; KASNECI, G.; WEIKUM, G. YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia. *WWW 2007*, 2007.

[7] LEHMANN, J. et al. *DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia*. [S.l.], 2012. v. 1, 1–5 p.

[8] MILLER, G. A. *WordNet: A Lexical Database for English George A. Miller*. 1995. 39–41 p.

[9] BOLLACKER, K. et al. *Freebase: A Collaboratively Created Graph Database For Structuring Human Knowledge*. [S.l.], 2008. 1247–1249 p.

[10] NICKEL, M. et al. *A Review of Relational Machine Learning for Knowledge Graphs*. [S.l.], 2015.

[11] GETOOR, L.; TASKAR, B. *Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning)*. [S.l.]: The MIT Press, 2007. ISBN 0262072882.

[12] LAO, N.; COHEN, W. W. Relational Retrieval Using a Combination of Path-Constrained Random Walks. *Machine Learning*, v. 81, n. n.1, p. 53–67, 2010.

[13] GARDNER, M.; MITCHELL, T. Efficient and Expressive Knowledge Base Completion Using Subgraph Feature Extraction. p. 1488–1498, 2015.

[14] BORDES, A. et al. Translating Embeddings for Modeling Multi-Relational Data. *Advances in NIPS*, 2013. ISSN 10495258.

[15] LIU, H.; WU, Y.; YANG, Y. Analogical Inference for Multi-relational Embeddings. 2017.

[16] MIKOLOV, T. et al. Efficient Estimation of Word Representations in Vector Space. 2013.

[17] ANDREWS, R.; DIEDERICH, J.; TICKLE, A. B. *Survey and critique of techniques for extracting rules from trained artificial neural networks*. [S.l.], 1995. 373–389 p.

CBLP: a case-based approach to predict human mobility

Silva, C. I. L. C. ; Brandão, A. A. F. Siqueira, F. L. †;
Laboratório de Técnicas Inteligentes - Escola Politécnica - USP
†Laboratório de Tecnologia de Software - Escola Politécnica - USP
E-mail: igortkd2006@gmail.com {anarosa.brandao, levy.siqueira}@usp.br

Abstract—Nowadays, there are several devices that can provide GPS information continuously. The availability of all this information combined with the improvement of techniques to analyze massive amount of geolocation data opens up new possibilities of services related to human mobility. Smart traffic control, safety planning routes, and smart and safe driving are among the domains that can benefit from models to predict human mobility. In this paper we present a new approach to predict human mobility based on Case Based Reasoning (CBR). Using the Geolife GPS dataset to apply and analyze our proposal, we obtained as result 72% of accuracy, a value similar to the obtained by other studies. Our conclusion is that CBR can be successfully applied to geolocation prediction. Moreover, there are still many improvements that can greatly improve its accuracy.

Keywords— geolocation prediction, case based reasoning, data mining, mobility big data.

Classification — Master's degree.

Category— Beginner

I. INTRODUCTION

The massive amount of geolocation personal data can provide useful information about user's mobility. Nowadays the user's historical location can be provided by devices with embedded geolocation position systems (GPS) such as smart phones, navigators, trackers, notebooks, wearables, vehicular navigation systems, and others [1]. The wide variety of devices and their ability to generate records in short intervals, 1 to 5 seconds [2] [3], [4] enables the collection of massive amount of geolocation data and introduces the concept of Mobility Big Data (MBD) [1].

Geolocation data can feed Geolocation Prediction Models (GPM) [1] in order to predict personal information about a next geolocation based on current geolocation. Moreover, the MBD analysis can provide meaningful inputs to the GPM, e.g. information about the region where the user is at the moment may have more value to the model than a GPS point (latitude, longitude and altitude). According to Xu et al [1], this combination of GPM and MBD can provide or improve several location-based services such as smart traffic control, safety planning routes, resources recommendations, smart and safe driving, social interaction, personalized notifications, among others.

In a recent survey, Xu et al. [1] categorize solutions to GPM associated with MBD as: (i) Markov-based methods; (ii) Bayesian network-based methods; (iii)Neural Networks-based methods; and (iv) Regression-based methods. The use of such approaches present an average accuracy that range from 39% to 78% [1]. Nevertheless, Lu et al [5] and Song et al [6] presented that human mobility is 93% and 95% predictable respectively, which means that there is still room to improve the average accuracy of GPM.

In this paper we propose CBLP, an CBR approach to estimate geolocation prediction. The CBLP results in a real life dataset indicates a promising method that should be more explored. This document is organized as follows: section II presents some background for understanding our proposal; section III some existing work on the theme; section IV our proposal and section V the application of our model to a dataset. In section VI we present the evaluation of our model and associated results and, finally, our conclusions and future work in section VII.

II. BACKGROUND

A. Mobility Big Data Analysis

Mobility Big Data analysis consists of processing and clustering geolocation raw data into popular geolocation regions (PGR), followed by mining personal trajectories (PT) from the PGR's [1]. In order to apply our approach, Geolife [4], a dataset containing mobility raw data, was chosen as the data source, and DBSCAN as the algorithm for clustering such data. Having the clusters, each PGR is represented by the cluster it belongs and PT are mined considering clusters as abstractions to their starting, middle and ending PGR's [1].

1) Mobility Dataset: Geolife is a GPS trajectory Dataset collected by Microsoft Research on Asia. It contains data from 182 users collected from 2007 to 2012 and describes more than 17 thousands trajectories. The Geolife database is composed of several files each one representing one or more trajectories on a specific day for a specific user. By trajectory we mean a sequence of records where the difference between two subsequent records is at most 300 meters or 30 minutes. Its files contain logs oflocation and time records. Each record is represented in a row and composed by seven fields: <latitude, longitude, 0, altitude, #daysfrom 12/30/1899, date, time>. The number of records in Geolife dataset is 23.42×10^6.

2) Clustering Algorithm: The density-based spatial clustering of applications with noise (DBSCAN) algorithm [7] is a clustering algorithm that groups points that are closely packed with a minimum near points and mark as noise points alone in a region with few near points. DBSCAN require two

parameters to be executed: EPS and MinPoints. EPS defines that a point *P* is part of a cluster *C*, whenever the distance from *P* to any existing point of *C* is equal or less than EPS. The MinPoints defines the density of the clusters and determine the minimum density of points necessary to form a cluster.

3) Mining Trajectories from Clusters: Mining a trajectory from clusters consists of reducing dimensionality by obtaining a personal trajectory (PT) based on popular geolocation region (PGR).Therefore, having a GPS dataset such as Geolife, mining a representation of a GPS trajectory to a PT basically consists of mapping the dataset geolocation records to a sequence of PGR where each geolocation record belongs. An example of a trajectory dimensionally reduced is: PT = $< ClusterA, ClusterB, ClusterA >$.

B. CBR: Case Based Reasoning

CBR is Artificial Intelligence approach to solve problems based on similar past cases [8]. A CBR system solves new problems (target problems) by adjusting or using solutions from old problems (analogue problems) [9].The CBR methodology have four main steps: (i) retrieve similar analogue cases to the target problem; (ii) reuse analogue solutions to propose a new solution to the target problem; (iii) revise if necessary to improve the solution of target problem; and (iv) retain the new solution to improve knowledge.

Some advantages of using CBR are: (i) only deal with real problems avoiding to worry about all possible problems; (ii) handle risk cases by identifying scenarios with high error degree; (iii) there is no need of having a good understanding about scenario domain because solutions are based on past cases data collected; (iii) large number of solution applied to different domain including recommendation systems.

III. Related Work

Ashbrook [10] used Markov models to calculate a matrix of probabilities of movements between locations. They collected the data used in experiments and they did not exposed average accuracy for all users.

Herder et al [11] implemented a method that use probabilistic functions, Markov methods and closest locations to the user current location to predict the next location. Their work applied techniques used to predict revisitations in websites to geolocation prediction. They state the existence of a strong dependence between destinations and period of the day and use it as an additional input to their method. Moreover, they adopted Geolife as the dataset and considered that each dataset file contains only one user trajectory. Such choice can hide important information, where an interval of more than 60 minutes may indicates the end and the beginning of two subsequent trajectories.

Khoroshevsky and Lerner [12] proposed a method based on probabilistic functions to compute the location prediction. Geolife was chosen as the dataset. Diversely from [11], they pre-processed the dataset to shrink it while maintaining the utility of information collected in its files. Therefore, they established a limit of 20 minutes and 50 meters between subsequent records in a file as a way to discard trajectory middle points. We advocate that by using such limits they may have discarded starting and ending points of trajectories, since a Geolife file may contain more than one trajectory in a file.

Gambs et al [13] used Markov method to predict if a user is at home, at work, or at other place. They use a probabilistic method based on user's last visitations to predict the next locations. Their approach is limited in respect to the possibilities of different destinations.

Different from methodologies cited above our approach strongly depends on finding similar cases to the target trajectory. Naturally, more data should result in more assertive choices for similar cases, but our model does not need to limit the amount of data necessary to execute the prediction. In other words our approach accuracy is not restricted by a big amount of data, but by the similarity of data.

IV. The Case-Based Location Prediction Model (CBLP)

The CBLP is based on CBR and it is composed of two steps extracted from CBR: (i) retrieve most similar analogue trajectories to a target trajectory; (ii) reuse analogue data to predict the target trajectory. Our model assumes that the most similar the trajectories are, the greater the likelihood of their destination be the same. These assumptions considers that human mobility is strongly influenced by past behavior [5]. An overview of our model is presented in figure 1.

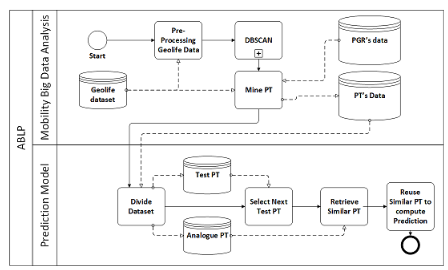


Fig. 1: Case Based Location Prediction (CBLP) Model.

A. Retrieve Similar Trajectories

In order to retrieve similar analogue trajectories it is necessary to define some distance to compare trajectories. Similarity is evaluated by calculating the distance between each analogue trajectory and the target one. The closest ones are considered similar to the target. We used the ED to measure the similarity. To properly use ED it is necessary to convert a trajectory into an n-dimensional array containing features with high relevance to the trajectory destination. According to [6], [14] in most cases human mobility are re-visitations with high temporal and spatial regularity, because Geolife is composed exactly with spatial and temporal data this understanding justifies the use Geolife of predict mobility. Focusing on comparing and retrieving similar trajectories is possible to infer several inputs from Geolife data such as the nearest buildings or places for spatial attributes and the day of week or period of the

TABLE I: Time features to numbers

Time feature	Symbol	Number
Period of the day	AM/PM	0/1
Day type	Week/Weekend Day	0/1
Day of the week	Mon/Tue/Wed/Thu/Fri/Sat/Sun	0/1/2/3/4/5/6

Source: Adapted

day for temporal attributes. All these features can be used to characterize a trajectory focusing on space and time.

We use latitude and longitude because the ED of geolocation attributes are proportional to the geographic distance, this means the closer to 0 is the ED between two trajectories' spatial attributes, the more geographic closer are the trajectories' points.

According to Herder, Siehndel and Kawase [11], trajectory destination (end point) is dependent on the time of day which the trajectory begins (start point) and it is also dependent on the type of day it occurs (week or weekend). Based on this understanding, we also defined the features (i) period of the day (AM or PM), (ii) day type (weekday or weekend day), and (iii) day of the week (Monday, Tuesday, ..., Sunday) for representing the time attributes which represent a trajectory. Both attributes (time and geolocation) refer to the starting point of a trajectory and were obtained at the moment of mining PT's. Then, we assign numerical values to each time feature, as indicates table I. Since time features and geolocation attributes are represented using different units, a normalization is necessary to ensure that each attribute has the same impact on the result.

An example of the normalized array is given by

$$< 0, 0.00854, 0.03342, 0.45609, 0.97231 >,$$

where the array entries represents, respectively, the period of the day; the day type; the day of the week; the latitude of the starting point; the longitude of the starting point. Having calculated the distance between each analogue trajectory to the target one, we consider the closest three as the similar analogue trajectories.

B. *Reuse Similar Trajectories*

The reuse of similar trajectories to derive the next destination considers the fact that they have similar features of time and space of the target trajectory by construction. Therefore, the algorithm has to choose, among the similar analogue trajectories, a single one to serve as the derived solution. Whenever possible, our algorithm chooses the most frequent (the mode) among the most similar analogues to the target trajectory, otherwise, it chooses the one most similar analogue.

V. Applying CBLP to Geolife

A. *Geolife Data Analysis*

In order to apply our approach to Geolife, we first need to conduct the MBD analysis to it and extract the PGR's and PT's for each user. These steps are applied to all dataset with no restrictions. We already defined on section II our understanding of what is a trajectory as well as the algorithm used to cluster points into PGR, the DBSCAN. To implement DBSCAN we use the Apache Machine Learn Library [15]. In addition, for the purpose of avoiding DBSCAN performance issues while running with large datasets, we decided to discard middle points of all trajectories, following the approach already adopted by [12]. By doing this, the number of records to be clustered by DBSCAN decrease from $23.42 * 10^6$ to $1.09 * 10^5$.

As aforementioned, the algorithm requires EPS and MinPoints as input parameters. Nevertheless, such values impacts the result of the DBSCAN execution depending on the number of records for a user. In order to mitigate the impact of the choice for EPS and MinPoints values in the algorithm accuracy, we ran DBSCAN for EPS ranging from 5 to 30, with a step of 5 units, for each user. Also, MinPoints are set to range from 1 to 4, because we do not want to restrict the creation of a new cluster. Since all points are departures or destinations, larger values of MinPoints may exclude locations that were visited just a few times.Therefore, all combinations for (Users, EPS, MinPoints) were considered to generate the dataset that serves as input for CBLP resulting in 24 (6 EPS * 4 MinPoints) experiments executed for each user.

Mining PT follows the approach described on section II-A3 and the resulting data serve as input for the Prediction Model of CBLP. A total of 62544 PT populate the PT dataset.Having the personal trajectories (PT) yielded by the Geolife data analysis, the next step is applying the Prediction Model to them. Therefore, retrieving and reusing similar trajectories are the steps to be conducted using the PT dataset.

B. *Retrieving similar trajectories*

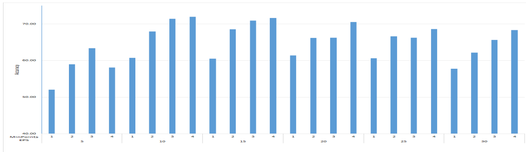
In order to prepare PT data to apply the prediction model, we split the resulting data from each user into two parts chronologically ordered: the first $\frac{2}{3}$ of PT were considered as analogue cases and the last $\frac{1}{3}$ of PT were considered as targets to be predicted. The retrieving process consists of, for each target PT, evaluating similarity between the target PT and all analogue PT and selecting the three most similar PT.

C. *Reusing similar trajectories and deriving destination*

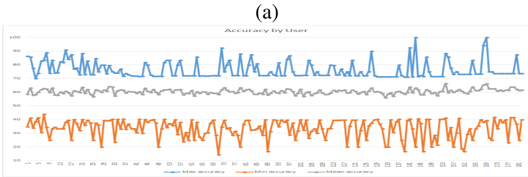
For each target, from the three most similar trajectories, we verify if there is overlap of destinations. If so, we chose as the target destination the most frequent one, otherwise we choose the target destination as the one from the most similar trajectory.

VI. Evaluation and Results

In order to evaluate our model we perform the steps described in section V for each of the 4368 (=182*6*4) combinations of users, EPS and MinPoints, performed in the time of 3 days. Therefore, executions with less than 10 personal trajectories (at least 7 considered as analogues and 3 as targets) were ignored to avoid trending data. Each one execution has a set of destinations to be predicted and the success of prediction will be given by the comparison of the predicted destination with the ones of the target PT dataset. Accuracy was calculated considering the ratio between the



(a)



(b)

Fig. 2: A - Accuracy due to EPS and MinPoints. B - Accuracy results performed by each user.

TABLE II: Prediction Accuracy Models

Author	Dataset	Locations per user	Accuracy
CBLP	Geolife (182 users)	12	0.72
Gambs et al., 2012	Geolife (175 users)	8	0.7 - 0.95
Khoroshevsky and Lerner[1]	Geolife (168 users)	4	0.82
Khoroshevsky and Lerner[2]	Geolife (168 users)	19	0.75

Source: Adapted from Khoroshevsky and Lerner.

number of successful prediction ($Succ$) and the total number of target trajectories ($Total$).

Figure 2 presents a plot with the results of all experiments performed. Each column represents the results of the average accuracy for all users, considering an specific value for EPS and MinPoints. By analyzing the plot, we may infer that the best values of (EPS, MinPoints) DBSCAN parameters, considering the accuracy ($> 70\%$), were (10,4), (15,4), (10,3) and (15,3). If we consider the our best combination for EPS and MinPoints, our accuracy is 72%. Nevertheless, since we execute all the possible combinations of (EPS,MinPoints) for each user, we also calculate the mean average accuracy for all of them, which results in 60.51%.Figure 2B presents the Max, the Average, and the Min results of accuracy for all DBSCAN parameters, with each point representing a user.

A comparison with existing work that used Geolife is presented in table II and as we see, our approach did not overcome [12] or [13], but some considerations are relevant. When comparing to [12], their best results improve ours by 10% and this difference can be justified by the difference in locations discovered by user. If the cluster process generates a few numbers of locations as possible destinations, it is reasonable that your accuracy will be improved because, statistically, there is a greater probability of success. When comparing with [13], it is important to remark that their results are displayed for 3 different datasets, but specifically for Geolife their accuracy ranges around 60% - 70% so CBLP overcome [13] specifically for Geolife.

VII. Conclusion

Accurate prediction of user's mobility can improve services based on geolocation. In this paper we proposed CBLP, an case-based model to predict the human mobility. Although CBLP does not overcome other approaches to the same problem, the accuracy calculated under the same basis is close of existing work that adopts the same dataset. This was our first incursion of using a CBR-based approach in such a domain and our choices were the simple ones, mainly to check if it could be promising or not.

Differently from others our method is not probabilistic avoiding restrictions about amount of data. Probabilistic methods like Markov or Bayesian are the most widely used according to Xu *et al*. To our understanding, encouraging new approaches is crucial to improve the field research. The results encourage us to advocate that CBLP was successfully applied in the human mobility and that there is room to improve it. In fact, Aamodt and Plaza [8] argue that the retrieving step of a CBR cycle could provide better results with the adoption of techniques such as nearest neighbor, induction, statistics, neural networks, fuzzy logic, etc. Improvements may also come from the inclusion of the revise and retain steps of the CBR cycle in the CBLP model.

References

[1] XU, G. et al. A survey for mobility big data analytics for geolocation prediction. *IEEE Wireless Communications*, IEEE, v. 24, n. 1, p. 111–119, 2017.

[2] WU, X. et al. Data mining with big data. *ieee transactions on knowledge and data engineering*, IEEE, v. 26, n. 1, p. 97–107, 2014.

[3] ZHENG, Y. et al. Understanding mobility based on gps data. In: ACM. *Proceedings of the 10th international conference on Ubiquitous computing*. [S.l.], 2008. p. 312–321.

[4] ZHENG, Y.; XIE, X.; MA, W.-Y. Geolife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.*, Citeseer, v. 33, n. 2, p. 32–39, 2010.

[5] LU, X. et al. Approaching the limit of predictability in human mobility. *Scientific reports*, Nature Publishing Group, v. 3, p. srep02923, 2013.

[6] SONG, C. et al. Limits of predictability in human mobility. *Science*, American Association for the Advancement of Science, v. 327, n. 5968, p. 1018–1021, 2010.

[7] ESTER, M. et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Kdd*. [S.l.: s.n.], 1996. v. 96, n. 34, p. 226–231.

[8] AAMODT, A.; PLAZA, E. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, IOS press, v. 7, n. 1, p. 39–59, 1994.

[9] WATSON, I. Case-based reasoning is a methodology not a technology. *Knowledge-based systems*, Elsevier, v. 12, n. 5-6, p. 303–308, 1999.

[10] ASHBROOK, D.; STARNER, T. Using gps to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous computing*, Springer, v. 7, n. 5, p. 275–286, 2003.

[11] HERDER, E.; SIEHNDEL, P.; KAWASE, R. Predicting user locations and trajectories. In: SPRINGER. *International Conference on User Modeling, Adaptation, and Personalization*. [S.l.], 2014. p. 86–97.

[12] KHOROSHEVSKY, F.; LERNER, B. Human mobility-pattern discovery and next-place prediction from gps data. In: SPRINGER. *IAPR Workshop on Multimodal Pattern Recognition of Social Signals in Human-Computer Interaction*. [S.l.], 2016. p. 24–35.

[13] GAMBS, S.; KILLIJIAN, M.-O.; CORTEZ, M. N. del P. Next place prediction using mobility markov chains. In: ACM. *Proceedings of the First Workshop on Measurement, Privacy, and Mobility*. [S.l.], 2012. p. 3.

[14] GONZALEZ, M. C.; HIDALGO, C. A.; BARABASI, A.-L. Understanding individual human mobility patterns. *nature*, Nature Publishing Group, v. 453, n. 7196, p. 779, 2008.

[15] APACHE, S. F. *Math – The Commons Math User Guide - Machine Learning*. Disponível em: <http://commons.apache.org/proper/commons-math/userguide/ml.html>.