ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.

A neuro-symbolic approach to GPS trajectory classification

Raul Barbosa[1], Douglas O. Cardoso[1,2], Diego Carvalho[3], Felipe M. G. França[1],

1-PESC/COPPE – UFRJ, Brazil
2-Coord. Eng. da Computação -PT, CEFET/RJ, Brazil
3-PPPRO, CEFET/RJ, Brazil

Abstract. This paper proposes approaches to GPS trajectory classification problem in the context of the Rio de Janeiro's public transit system (with hundreds or more classes). We adopt a weightless neural network architecture combined with both spatial partition and multiclass decision graphs, inspired by a neuro-symbolic sense of adding knowledge from the domain as opposed to the use of a raw machine learning. Experimental results show performance boosts when using some of the proposed strategies.

1 Introduction

In tandem with the current advances in technology, a profusion of devices are shipped with geolocation capability everyday, providing data about their real time location through global navigation satellite system such as the Global Positioning Sytems (GPS). The generated location data have been used on a pleiad of applications such as taxi fraud detection [1], behavioral identification and classification [2], among many others. This work explores another application of GPS data known as trajectory classification as posed by [3].

Our work aims to enhance the operation of a large city as Rio de Janeiro, where 3.7 million of people commute every day by bus. Our contribution aims to use the geolocation data provided by the city's buses to pinpoint deviations from standard operation. Besides, we believe that our framework can be generalized for other geospatial and non-geospatial classification problems.

In this context, we propose and evaluate three approaches to identify a bus route based on its GPS data based on preexisting digitized trajectories of buses. We implemented a lightweight classification architecture known as the WiSARD to help us deal with the data deluge. The WiSARD model has been used in many different previous works and contexts (see [4] and [5]), always with competitive running time, accuracy and low standard deviation.

The remainder of this work is divided in four sections as follow: Section II describes the Rio de Janeiro's public transit and its geographical tracking system. Section III describes our approach to the problem based on a weightless neural network architecture. Section IV depicts a set of experiments with real data. Each experiment is based on a different image building paradigm. Besides, this section presents the results and shows how we can benefit from using different grid strategies. Section V wraps up and finishes the work, giving some ideas and future thoughts.

2 Rio's buses GPS trajectories

The current legal structure of the Rio de Janeiro's transit bus system back to the year of 2010, when a set of regulations imposed the real time geolocation on every vehicle. The data provided by the Rio's prefecture was composed by a timestamp, vehicle id, route tag, latitude, longitude and velocity for every vehicle in the system.

The use of geolocation data seems straightforward, nevertheless this idea is misleading due to its incredible amount of noise. A preliminary analysis show that out of 237,145,699 GPS measures, more than 25% would have been discarded by city managers because the lack of a route tag. Besides, drawing vehicles trajectories on a map shows a clear evidence that some vehicles are labeled with one route tag but actually serving another. In order to use the available data to cross-check the current level of fulfillment of Rio's transit regulations, we used a classifier to re-tag the trajectories, empowering the city management with more reliable data.

3 The WiSARD weightless neural network architecture

The WiSARD model [6] is a weightless neural network originally developed as a hardware architecture for image recognition [7]. This kind of neural network works with basic read and write operations in RAM memories. WiSARD is mainly organized in *discriminators* which are groups of RAMs (also called neurons), being one discriminator frequently used to determine just one class of many given as input to the classifier.

The standard WiSARD model can be categorized as a supervised learning method. During the training phase, a number of binary vectors (examples) are shown to the classifier. Each input vector is mapped to a *retina*, which is a division of the input vector in K tuples of N bits, commonly in a pseudo-random manner. Following this strategy, each N bit tuple is used as an address of a RAM memory of 2^N positions, with the K tuples mapping K memories.

Learning from an example is just as simple as writing a '1' in all memory addresses mapped by a single input. Classification is handled by first mapping an input in the same way training examples were. Afterwards, we present the input to all discriminators. The score for a discriminator 'y' is calculated by counting the number of memories mapped by the classification example whose addresses were previously written by any training example. The discriminator that has the greatest score is chosen as the output class. For a visual description of the architecture, the reader can refer to Figure 1 and [6, 7].

4 A weightless neuro-symbolic classsifier

Since the WiSARD model works with a binary vector as its input unit, a common step towards its use consists of a pre-processing step (like any other classifier would require). We propose three main approaches for generating our input.

4.1 Generating an initial image

One straightforward approach to converting our GPS data to a binary vector is to generate an binary image out of the GPS data, composed by every GPS

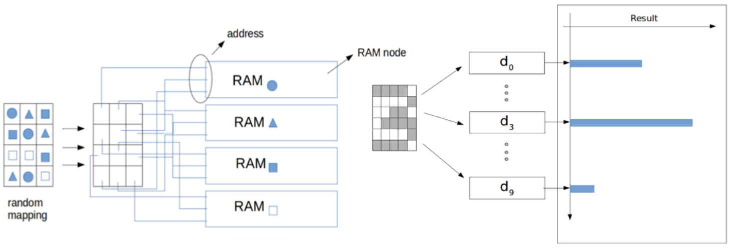
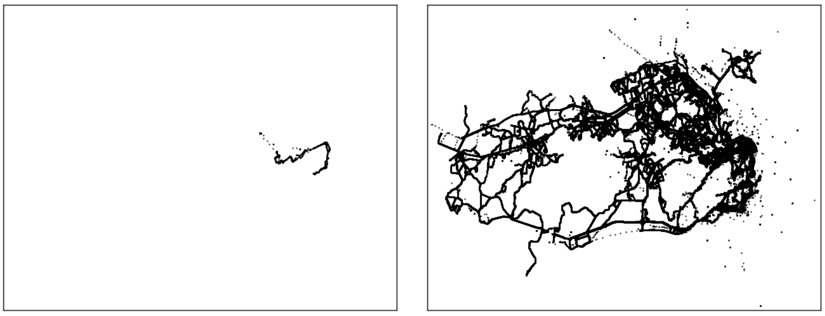


Fig. 1: An example of a discriminator (left) being trained. The retina is divided into tuples that become addresses of the RAMs. Afterwards (right), the classification is done by presenting an input to the network and for all trained classes (discriminators), the one with the best result is chosen.



(a) The footprint of a vehicle. (b) The footprint of the entire fleet.

Fig. 2: Examples of the problem data and its usage.

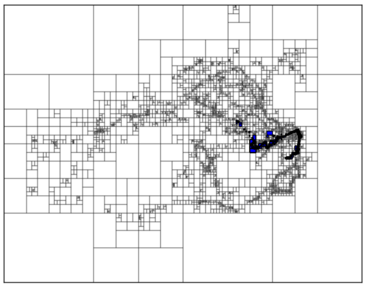
coordinate as points in a 2-D canvas, as a footstep or footprints, within some reasonable limits as considered by [8]. This approach tackles a brief loss of GPS data as some noisy example among other good ones. Also it collapses similar GPS addresses, giving priority to a complete trajectory image. Figure 2a shows a single trajectory – footprint – performed by a vehicle that served only one route during a period of time. In order to understand the complexity of the classification task, Figure 2b presents the trajectories of all vehicles, serving all lines during the same time period of time. The footprint of a vehicle is our unit of classification, representing an example of a bus route for our classifier.

4.2 A smart gridded image

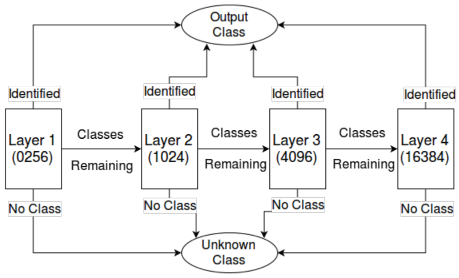
On top of the previous approach, we also know that we have some prior knowledge about the nature of the data that we are working with. As figure 2b shows, taking a small look at the data distribution over the 2-D space can show some regions of interest.

We then propose to transform our initial grid into a more symbolic representation, readjusting its cells sizes. We do it guided by a kd-tree [9] division of the space, limited by a threshold in the number of leaves and the size of pixel. The

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1. Available from http://www.i6doc.com/en/.



(a) A footprint example after the grid preprocessing.



(b) The decision directed acyclic graph approach towards an ensemble classifier.

Fig. 3: Example of the gridded input and description of the DDAG proposal.

kd-tree division is given by different strategies of splitting a leaf (at a median point, for example) and deciding the next leaf/rectangle to be divided. According to this representation, figure 3a shows an example of a grid and one input when using this approach. This approach can be thought as an enhancement of the previous approach, with some sort of region-based clustering (as seen in [2]) for feature extraction.

4.3 A DDAG ensemble decision based tree model

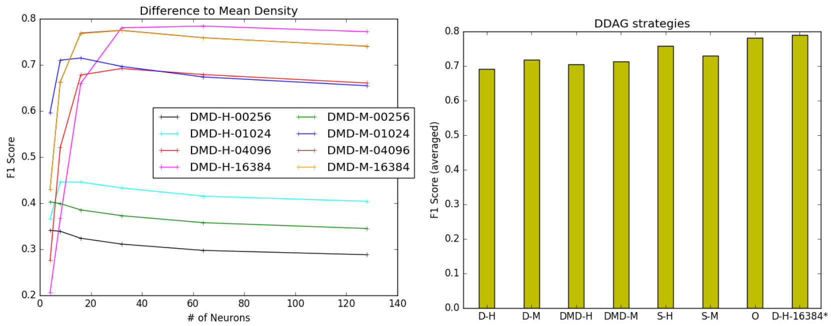
Going further on the neuro-symbolic path, our third proposed approach is an ensemble of WiSARD classifiers, arranged in a decision directed acyclic graph - DDAG [10], a commonly used strategy for multiclass problems. The main goal here is to propose another way of breaking ties, a common and well-known problem of WiSARD models, eliminate spurious example and aggregate more prior knowledge to better classify.

A less detailed WiSARD (small retina) is trained and used to initially classify an example. Following this classifier, different WiSARDs, each one with a more detailed grid, are trained and used to classify the example. The process stops when no class can be identified, only one class remains or the final level is reached, with its output being the model‘s response. Each graph node can be thought of a step of classification using the second approach but with less classes to choose from. Figure 3b tries to put this strategy into picture.

5 Experiments and results

We used a month of data (between June and July of 2015) as our test data in order to evaluate our models. This data was then pre-processed, resulting in images of 620x480 pixels. This dataset has a size of almost 50GB involving the classification of more than 400 classes.

Regarding the second approach (Section 4.2), we tried two splitting methods – dividing a node at the median (M) and in half (H) – and three heuristics to decide the next node to be divided - largest size of node or how many points



(a) Results using absolute difference to mean density (of points) of a leaf as criteria for choosing the next node split.

(b) Results for the DDAG approach, as well the best kd-tree grid(D-H-16384*) and the plain full grid approach (O)

Fig. 4: Results of kd-tree grid and DDAG ensemble.

Solution	Grid Size	# Neurons	F1 score (mean)	F1 Score (std. deviation)	Avg. Execution Time (s)
Original Grid	297600 (620x420)	64	0.7815	0.0005	1375.98
Best KDTree Grid (Density/Half)	16384	32	0.7896	0.0009	602.65
2nd Best KDTree Grid (Difference to Mean Density/Half)	16384	64	0.7843	0.0010	2641.93
Best DDAG (Size/Half)	256/1024/4096/16384	4/16/32/64	0.7590	0.0006	1537.57

Table 1: Summary of strategies and their best results.

inside a node ("Size",S), greatest density of a node ("Density",D), and largest absolute difference to the mean density of the current node set ("Difference to Mean Density", DMD). Also, we tried four different grid sizes, resulting in datasets of size up to 4GB.

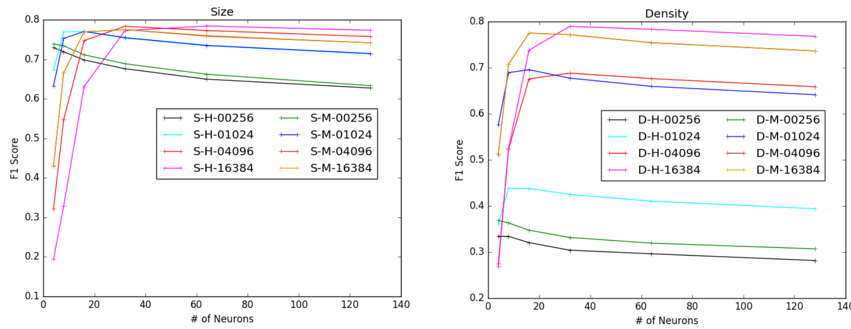
Finally, to evaluate the last model, we decided to take the 4 levels of classifiers (one for each grid size we tested before). The number of neurons used during WiSARD model design was also tested, varying from 4 to 128 neurons 'K' (except for the evaluation of the DDAG-model, when we fixed this number to the best value for each grid size in the second approach). It is important to note that the number of neurons given a fixed retina fixes also the number of addresses.

We report in figures 4 and 5 the results of 5 rounds of ten-fold cross-validation, summarized by the f1 measure. The f1-measure was chosen due to a high unbalanced training set, as a mean of both precision and recall. Table 1 highlights the winners for each approach with both training and classification time (excluding initial I/O and pre-processing steps).

6 Conclusion

This paper showed a framework to deal with real time GPS data classification using the WiSARD classifier. The proposed kd-tree grid was able to achieve the best performance, indicating that other tree structures may also be useful. The presented framework can be an alternative when dealing with large data masses and we believe that it can be used to solve other similar problems such

ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Bruges (Belgium), 26-28 April 2017, i6doc.com publ., ISBN 978-287587039-1.
Available from http://www.i6doc.com/en/.



(a) Results using greatest size of leaf as criteria for choosing the next node split.

(b) Results using greatest density of a leaf as criteria for choosing the next node split.

Fig. 5: Plot showing the results of kd-tree grids.

as rogue vehicle detection. The WiSARD architecture made this work feasible, due to the large amount of classes involved in the problem. We still need to investigate more griding strategies and compare with other baseline classifiers and commonly used datasets. Current work is focusing on the improvement of the DDAG's creation.

References

[1] Siyuan Liu, Lionel M Ni, and Ramayya Krishnan. Fraud detection from taxis' driving behaviors. *IEEE Transactions on Vehicular Technology*, 63(1):464–472, 2014.

[2] Jae gil Lee, Jiawei Han, Xiaolei Li, and Hector Gonzalez. Traclass: Trajectory classification using hierarchical region-based and trajectory-based clustering, 2008.

[3] Yu Zheng. Trajectory data mining: An overview. *ACM Trans. Intell. Syst. Technol.*, 6(3):29:1–29:41, May 2015.

[4] H. C. C. Carneiro, F. M. G. França, and P. M. V. Lima. Multilingual part-of-speech tagging with weightless neural networks. *Neural Networks*, 66:11–21, 2015.

[5] Rafael Lima De Carvalho, Danilo SC Carvalho, Félix Mora-Camino, Priscila VM Lima, and Felipe MG França. Online tracking of multiple objects using wisard. In *ESANN 2014, 22st European Symposium on Artificial Neural Networks, Computational Intelligence And Machine Learning*, pages pp–541, 2014.

[6] Igor Aleksander, WV Thomas, and PA Bowden. Wisard a radical step forward in image recognition. *Sensor review*, 4(3):120–124, 1984.

[7] FMG França, M De Gregorio, PMV Lima, and WR de Oliveira. Advances in weightless neural systems. In *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2012.

[8] Diego Fonseca Pereira de Souza. Time-series classification with kernelcanvas and wisard. Master's thesis, Universidade Federal do Rio de Janeiro, 2015.

[9] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Communications of the ACM*, 18(9):509–517, 1975.

[10] John C Platt, Nello Cristianini, and John Shawe-Taylor. Large margin dags for multiclass classification. In *nips*, volume 12, pages 547–553, 1999.