On the Quality of Compositional Prediction for Prospective Analytics on Graphs

Gauthier Lyan, David Gross-Amblard, Jean-Marc Jézéquel

On the Quality of Compositional Prediction for Prospective Analytics on Graphs

Gauthier LYAN[1,2], David GROSS AMBLARD[2], and Jean-Marc JEZEQUEL[2]

[1] Keolis Rennes, FRANCE
[2] Univ Rennes, CNRS, Irisa Lab, FRANCE
gauthier.lyan@keolis.com, david.gross_amblard@irisa.fr, jean-marc.jezequel@irisa.fr

Abstract. Recently, micro-learning has been successfully applied to various scenarios, such as graph optimization (e.g. power grid management). In these approaches, ad-hoc models of local data are built instead of one large model on the overall data set. Micro-learning is typically useful for incremental, what-if/prospective scenarios, where one has to perform step-by-step decisions based on local properties. A common feature of these applications is that the predicted properties (such as speed of a bus line) are compositions of smaller parts (e.g. the speed on each bus inter-stations along the line). But little is known about the quality of such predictions when generalized at a larger scale.
In this paper we propose a generic technique that embeds machine-learning for graph-based compositional prediction, that allows 1) the prediction of the behaviour of composite objects, based on the predictions of their sub-parts and appropriate composition rules, and 2) the production of rich prospective analytics scenarios, where new objects never observed before can be predicted based on their simpler parts. We show that the quality of such predictions compete with macro-learning ones, while enabling prospective scenarios. We assess our work on a real size, operational bus network data set.

Keywords: Compositional · Machine learning · Graph · Prospective · Link Weight Prediction

1 Introduction

Machine learning techniques have been applied on static, graph-based applications, such as transportation networks or energy grids, to name a few [20,2,21]. These applications have in common an a priori topological model, i.e. a graph, where practical measures are performed on nodes and edges. In a a bus transportation network for example, it is possible nowadays to predict the awaiting time in a station or the probable duration of a trip with an acceptable accuracy [24,23].

Recently, micro-learning approaches have been successfully applied to more dynamic, prospective and what-if scenarios (e.g. power grid management[22]). In these approaches, ad hoc models of local data are built instead of one large model on the overall data set. Micro-learning is typically useful for incremental, prospective scenarios, where one has to perform step-by-step decisions based on local properties. Going back to our bus transportation example, a typical strategical prospective scenario applies when one bus network operator has to plan a new bus line that travels through road sections never used by the bus network until then.A common feature of these graph-based applications is that the predicted properties (such as speed/travel time of a bus line) are compositions of smaller parts (e.g. the speed or travel time on each bus inter-stations along the line). But up to our knowledge, little has been written about the quality of such composite predictions using micro-models, when generalized at a larger scale.

In this paper we propose a generic technique, graph-based compositional prediction, that allows 1) the prediction of the behaviour of composite objects, based on the predictions of their sub-parts and appropriate

composition rules, and 2) the production of rich prospective analytics scenarios, where new objects never observed before can be predicted based on their simpler parts. We show that the quality of such predictions compete with macro-learning ones, while enabling prospective scenarios. We assess our work on a real size, operational bus network data set.

The rest of the paper in organized as follows. In Section 2 we present our motivational scenario. Our model is outlined in Section 3. Section 4 shows our experiments. In Section 5 we discuss the related work, and present our conclusion and future work in Section 6.

2 Data Model and Motivation Scenario: Rennes City Bus Transportation

In public bus transportation networks, bus speed is considered to be a Key Performance Indicator (KPI) that translates the level of efficacy and attractiveness of a bus network [9,8,7] Consequently, bus networks operators struggle daily to maintain high bus speed and are in need of reliable, complete and flexible prospective methods to predict the performance (such as speed) of future bus lines.

Running Example. Our running example is the bus network of the French city of Rennes, forming a directed graph $G = (V, E)$, where V is the set of bus stops and $E \subseteq V \times V$ is the set of possibles one-stop trips. Such a graph can be considered as a static graph as long as most of its structure does not evolve in the short term (few months to few years). However, its edges generate a lot of data over time. We consider a non-empty set of features F, associated to each edge, with their corresponding types T_F. A typical set of features associated to each element of E is $F = (time, line, length, road - type, bicycle)$, where $time$ is a timestamp, $line$ is an integer denoting a bus line number, $length$ is the length of the inter-station section, $road - type$ indicates the kind of road the bus runs on (dedicated road or not), and $bicycle$ indicates whether bicycles are allowed. Figure 1 gives an overview of a bus network graph in which green vertices are departure terminals, blue vertices are transition/departure bus stops, white vertices are transitional bus stops and red vertices are ending terminals.

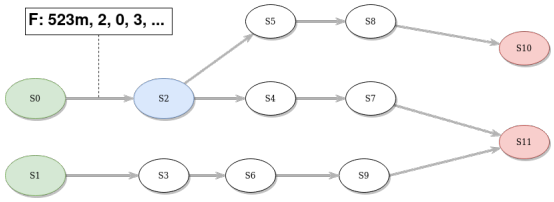

Fig. 1: A bus network with featured edges (F: length of the road, type of line, etc.)

Let us consider a measure $\mathcal{M}$ of the graph, defined on paths p in G. Let $\mathcal{D}$ be a learning data set of examples of $\mathcal{M}$. Let us consider a decision problem in the graph, e.g. Testing different configurations for a future bus line in the network. Given a source and target vertex:

Given a set of paths (i.e different configurations of the future bus line) S_p, for each path p in S_p from source to target (here $|S_p|$=1) predict the speed for every sub-section of p. This yields a dataset for every configuration, making it possible for the operator to evaluate which configuration is best suited for the future bus line depending on the speed at either inter-station level, section level, or bus line level.

In the next section we detail the corresponding learning problem associated with the prospective scenario.

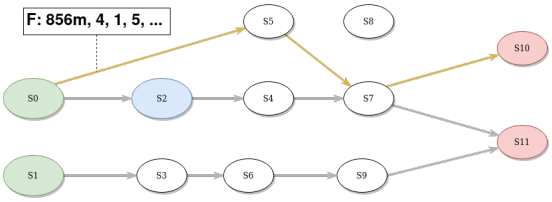

Fig. 2: Prospective scenario: predict $S0 - S10$ while edges $S0 - S5$, $S5 - S7$ and $S7 - S10$ do not exist in the data set

3 Micro-learning and Compositional Prediction

As a possible way to envision the prospective scenario, we propose the notion of compositional prediction using micro-learning:

- (Micro-learning) We will first learn a model of $\mathcal{M}$ on each individual edge of G.
- (Compositional prediction) Then we will build the model of $\mathcal{M}$ on a new path by composition.

More precisely, let $p = (e_1, \ldots, e_n)$ be a path in the graph G. Our goal is to predict a given measure $\mathcal{M}(p)$ on any path p on this graph. We possess local measures $m(e)$ of $\mathcal{M}$ on any edge e along this path and, typically, a composition law $\mathcal{C}$ links these measures:

$$\mathcal{M}(p) = \mathcal{C}(m(e_1), \ldots, m(e_n)).$$

Running Example. Consider that $\mathcal{M}$ is the trip duration for a precise bus and date, from the start to the end of the line (the path p). We have observations for $m(e)$, the duration of inter-station trips on p for this bus. Of course in this simple case the value of $\mathcal{M}(p)$ can be obtained by summing durations of trip duration $m(e_i)$ on all edges e_i of p. Similarly, the average speed could be obtained with a slightly different composition law, weighting duration by distances. Besides additive composition laws, multiplicative laws could be foreseen, such as the probability of bus failure (which is a product of one minus the bus failure probability on each edge).

We then define our compositional prediction approach:

Definition 1 (Compositional prediction). *Let $G = (V, E)$ be a directed graph, $\mathcal{M}$ a target measure, a composition law $\mathcal{C}$ and a data set $\mathcal{D}$ of observations $m(e)$ on edges of G. Given a target path $p = (e_1, \ldots, e_n)$, the compositional prediction of $\mathcal{M}(p)$ is given by:*

- *building a model $m^*(e)$ to predict $m(e)$ for each edge $e \in V \times V$, according to edge features $F(e)$;*
- *approximating the prediction of $\mathcal{M}(p)$ by*

$$\mathcal{M}^*(p) = \mathcal{C}(m^*(e_1), \ldots, m^*(e_n)).$$

Running Example. Suppose that we want to estimate the end-to-end duration of trip for a new bus line in a city. Given its path p, we would learn the typical duration of existing bus lines on each inter-stations road fragments (i.e. edges), and simply sum these predictions (the composition law). If a road fragment has never been visited by any other bus in our observation data set, the model approximates it using the most similar road fragment (according to its features such as road type, number of traffic lights, and so on.)

The main advantage of this approach is that predictions are based on the underlying structure of the graph, rather that on macro-observations on it. As shown in the examples, it yields a large flexibility in predictions, without domain specific knowledge (e.g., in the bus network context, multi-agent simulation requires complex modeling hand-made tuning [15]). But this method can also have its drawbacks. When an edge measure is missing, a strategy has to be deployed to fill the missing data, and this is highly related to the amount of describing features on the underlying graph. Also, the cost of building many models for each edge could be tremendous, regarding its macro-level counterpart. Finally, and more importantly, as each model bears its own approximations, errors may sum up along the compositional law $\mathcal{C}$, giving a potentially unusable prediction. We then identify the following research question: **Does micro-learning yield usable predictions with the compositional prediction strategy for prospective scenarios, and does it compete with traditional prediction methods (that apply in different contexts)?**

In the following experiments, we evaluate and discuss this question thanks to real data coming from a real bus network (thereby imperfect data, as this is most always the case in the real world [14,19]).

4 Experiments

4.1 Experimental settings

Reproducibility All of the experiments are reproducible with code and resources available on a dedicated repository[3].

Experimental environment All experiments were run on a 16 cores 32 threads @ 3500MHz, 64GB DDR4 3200MHz 1TB SSD NVMe, running Ubuntu 20.04 (Budgie flavour). Data were gathered into a normalized data lake built upon Apache Spark 2.4.5/Scala 2.11.12 jobs. All learning tasks were performed using GraalVM 20.2 and Scala SMILE 2.5.3. It took around 70 minutes for overall model training and 2 minutes for each of the following predictions.

Experiment with real data We considered the bus transportation system of the French city of Rennes and its metropolitan area (about 500,000 inhabitants) managed by the Keolis Rennes company. Our graph is the graph of 2110 bus stops and 119 bus lines[4], and the features of road portions are automatically extracted from OpenStreetMap (OSM in the sequel).

More specifically, the raw historical data set is composed of readings of bus travels between two contiguous bus stops. Table 1 shows a sample of this data set. The major columns are *line_type* (0 being high frequency, 1 urban, 2 metropolitan and 3 express metropolitan lines); *dwell_time*, the time passed at the origin stop (taking and unboarding passengers) and *speed*, the arithmetic speed between *orig* and *dest*. A single trip, which consists of all the measures of a single bus passing through every single bus stop of a bus line, can be identified by combining the columns *trip_start* which is the instant at which the bus has left the first point of the bus line, *sm* which is the service identifier, *chaining* which is the version of the bus line and *dir* which is the line direction. We considered only real data: no imputation was made in our data set. We considered only full trips for which all data points were present. We also filtered historical data used to feed the model, dropping invalid speeds/nulls/values and outliers. Using these rules, **we gathered 12 month of data between January 2019 and December 2019, for a total of around 15 millions tuples[5].**

[3] https://gitlab.inria.fr/glyan/compred

[4] Rennes transportation network: https://data.rennesmetropole.fr

[5] Data and more material is available at https://gitlab.inria.fr/glyan/compred

Prediction task We targeted the prediction of **the speed of a bus line** depending on time (holidays period, day, time of the day) and line type (urban bus line, metropolitan bus line). We considered predictions for existing lines (classical prediction) and non-existing lines (prospective scenarios, where the lines and some inter-stations data are absent in the training data set).

Candidates As experimental candidates, we chose 4 groups of bus lines for which we could gather data all along their path. We tested 13 different bus lines in total. The four groups contain different class of bus lines: urbans, inter-district, metropolitans and express.The urban group is composed of urbans bus lines, that cross the city center along their path. The inter-district group are urban bus lines that link different districts, avoiding the city center. The metropolitan group is composed of metropolitan bus lines, that links cities of the metropolitan area to the main city. Those lines contain urban, peri-urban and metropolitan sections. Finally, the express group contain metropolitan lines that allows a faster travel from and to external cities by servicing less bus stops. Note that in the network, each line share some sub-paths with others[6].

trip start	line	line type	sm	dir	orig	dest	speed	dwell time	order orig	order dest	section	chaining
2018-07-02 00:05:00	02	0	0221	A	2844	2842	25.8	0	1	2	1	21
2018-07-02 00:05:00	02	0	0221	A	2842	2804	21.0	23	2	3	1	21

Table 1: Sample of the historical data set

Road features with OpenStreetMap We have built a tool to look for a bus network in OpenStreetMap and to extract its infrastructure information, road per road. We then aggregate the data grouping the roads by inter-stations. We then sum the road lengths, number of traffic signals, stops, giveways, roundabouts, level crossings, crossings, and add the average legal speed of the roads.

Time discretization We categorized the time periods into six categories: No holidays, and Autumn/ Christmas/Winter/ Spring/ Summer holidays. For each category we divided each day coded from 0 (monday) to 6 (sunday) in 1 to 3-hours period (e.g. 1am-3am, or 7am-8am). This division is due to obvious pace changes in bus usage, as confirmed by the Keolis company.

Micro-learning data set The micro learning data set is made of measures for each inter-station of the network. We simply identified every distinct inter-station and added meta data such as holidays, bus line type, OSM data (that is constant through time), period of time in day and week day[7].

Macro-learning data set The macro learning data set is built using the raw data set, from which we extract **only the overall speed of every complete trip** (i.e trips for which we have data for each inter-station) of every bus line. We then add the bus line, its type and finally the OSM data aggregated all along the line (i.e total number of traffic signals, stops, etc.).

Learning method and validation methodology Our models are built using a classical random forest algorithm parameterized for regression and optimized using grid-search (this choice is motivated by its generality and overall performance, after selecting it amongst others: Lasso, OLS ,SVR, Cart, bayesian

[6] More material about the bus lines is available in the README of the repository
[7] More information about training data in the repository

ridge and gradient boosting, testing their performance using a small sample of data. It could be naturally adapted to more specific methods, but this is not the scope of our paper). For micro-learning, we build a model of inter-station speed. We then predict the speed of any trip by the natural compositional prediction $\mathcal{C}_s$ (summing the road lengths divided by the summed travel times, obtained with the predicted speeds). For macro-learning, we build a model of full bus-line speeds (from start to stop). Learning is performed with OpenStreetMap features, and without OpenStreetMap features on every bus lines in order to assess the impact of these features on the model precision. For validation purposes, we have removed all the data of candidates bus lines from the training data set of each model. To evaluate the classical scenario, we predicted the speed of an entire line with macro and compositional prediction. For the prospective scenario, we measured the compositional prediction quality on non-existing paths in the data set.

In the next sections we are coming back to our initial questions from Section 3.

4.2 Results

Accuracy on real data We tested (predicted) ∼12'500 trips for the urban bus lines group, ∼1'250 trips for the inter-district group, ∼5'800 for the metropolitan group and ∼800 trips for the express group. All those trips are distributed over the 6 holidays regime, 7 days and 17 periods in day over the 2019 year.

Figure 3 shows the predicted speed of one of the urban bus lines using micro-learning. Each point represents the average predicted speed (purple) and average real speed (dashed) of each inter-station of the line on all the time periods for which data was available. The leftest point of each curve represents the beginning of the bus line, the rightest one being the last stop. Observe that due to the bus line's data deletion from the training dataset, some inter-stations that are specific to this line are totally unknown to the model (orange dots). We also considered metropolitan, express and inter-district bus lines (not displayed)[8]. Visually at first, we can see that micro-learning captures well the daily behaviour of buses, even when the inter-stations are unknown to the model (orange dots).

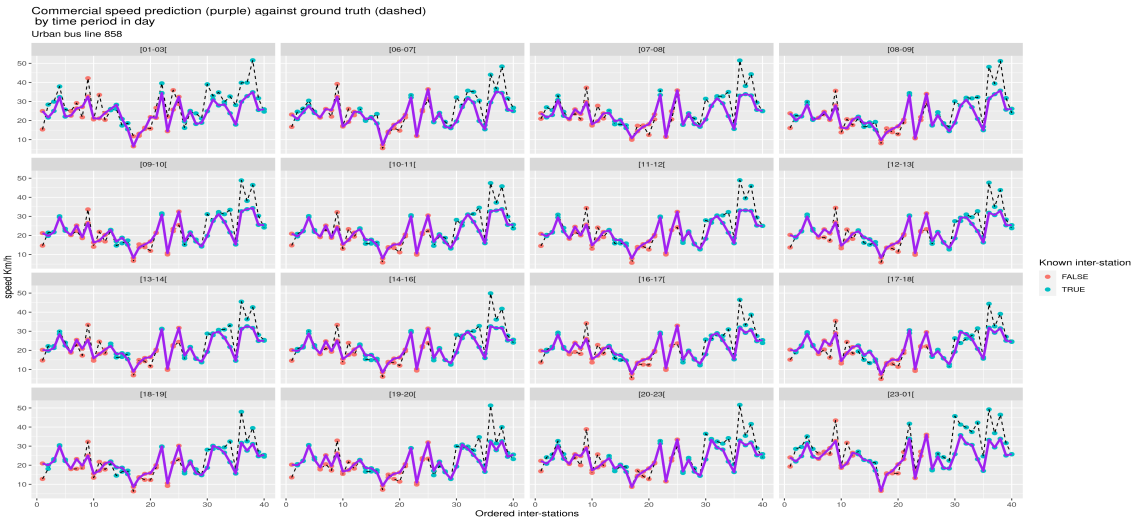

Fig. 3: Speed of urban bus line 858 at inter-stations using micro-learning (purple) vs true speed (dashed). Orange inter-stations where not used by any other bus, i.e. are absent from the training set.

[8] More material is available in the public deposit

Table 2 sums up the models's prediction errors for each bus line, group of lines (urban, inter-district, metropolitan and express) and all the lines (global). For each model and bus line (resp. group of lines), the table presents the Root Mean Squared Error (RMSE), the Mean Absolute Error (MAE) and the Mean Absolute Percentage Error (MAPE). RMSE emphasizes large residuals (outliers) while MAE and MAPE are less sensitive to residuals and easier to interpret [18]. RMSE and MAE unit is km/h while MAPE's unit is percentage (difference against truth in percents).

Table 2: Results table

Line	Micro known			Micro unknown			Micro global			Compred			Macro		
	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE
Urban lines															
858	7.3	5.0	18.2%	6.1	4.5	25.9%	6.9	4.8	22.4%	1.6	1.2	6.2%	2.1	2.0	8.4%
787	5.6	3.9	19.9%	9.1	7.3	44.4%	6.0	4.2	22.2%	1.7	1.3	7.2%	2.2	2.0	9.7%
785	5.9	4.5	22.7%	8.3	6.0	29.2%	7.5	5.4	26.6%	2.6	2.0	9.9%	2.7	2.5	10.4%
889	5.6	4.3	31.1%	8.7	5.9	25.4%	6.3	4.6	30.1%	3.0	2.7	17.4%	2.7	2.6	15.4%
689	6.3	4.6	29.4%	8.2	6.7	53.9%	6.4	4.7	30.7%	3.2	2.8	16.5%	2.4	2.3	12.2%
All urbans	6.3	4.6	25.2%	7.5	5.4	29.4%	6.7	4.8	26.5%	2.6	2.1	12.0%	2.4	2.3	11.5%
Inter-District lines															
696	8.9	6.4	27.1%	9.3	7.4	46.2%	9.0	6.7	33.5%	3.2	2.8	13.7%	4.0	3.8	15.5%
581	6.4	5.1	26.7%	10.3	7.6	31.8%	7.6	5.7	27.9%	3.5	2.8	12.3%	3.9	3.5	13.4%
All inter-Districts	8.8	6.4	27.0%	9.3	7.4	45.9%	9.0	6.7	33.3%	3.2	2.8	13.7%	4.0	3.7	15.5%
Metropolitan lines															
683	5.0	3.8	15.5%	8.4	6.7	26.9%	7.1	5.4	21.8%	3.2	2.5	10.3%	3.1	2.9	9.5%
849	6.3	4.7	37.4%	5.0	4.0	11.9%	6.3	4.7	36.6%	3.2	2.8	12.1%	3.7	3.5	14.3%
532	7.9	5.7	24.2%	12.1	11.9	427%	8.2	6.0	45.4%	4.0	3.3	13.7%	4.8	4.5	15.5%
969	7.2	4.8	25.3%	6.0	4.4	16.1%	7.1	4.8	24.4%	3.6	2.6	9.8%	3.5	3.2	9.9%
All metropolitans	6.7	4.8	25.6%	8.6	6.8	48.3%	7.3	5.4	31.7%	3.5	2.8	11.4%	3.8	3.5	11.9%
Express lines															
711	5.8	4.2	36.6%	4.5	3.6	10.1%	5.6	4.1	32.2%	3.6	2.7	10.3%	7.0	6.9	21.1%
859	6.1	4.2	27.8%	7.9	5.5	21.9%	6.6	4.6	26.1%	7.3	5.6	24.7%	7.0	6.8	20.2%
All express	5.9	4.2	33.4%	6.5	4.6	16.4%	6.1	4.3	29.8%	5.3	3.8	15.6%	7.0	6.8	20.8%
All lines															
All lines	6.6	4.7	25.5%	5.7	7.8	33.7%	7.0	5.0	28.0%	3.0	2.4	12.1%	3.2	2.9	12.2%

group	Compred standard dev.	macro standard dev.
urban	1.0	0.5
inter-district	0.7	0.3
metropolitan	0.9	0.4
express	0.9	0.0
global	0.9	0.4

Table 3: Models's sensitivity to time features

If we take a look at micro predictions in Table 2, we observe that prediction errors for known inter-stations (classical prediction) are better than prediction errors of unknown inter-stations (prospective predictions), except for the express bus lines group. Indeed, RMSE, MAE and MAPE can vary from a 1.2 to 2 factor between micro known and unknown. This shows that the features set used for micro learning might need some enhancement to help the model predicting better. Finally, we observe that global micro predictions precision vary between bus lines groups. However, the urban group gets the best predictions in all aspects when compared to other groups (RMSE:6.7, MAE:4.8, MAPE:26.5%), followed by express (RMSE:6.1, MAE:4.3, MAPE:29.8%), metropolitan (RMSE:7.3, MAE:5.4, MAPE:31.7%) and inter-district (RMSE:9.0, MAE:6.7, MAPE:33.3%) groups.

Urban bus lines are the major lines in the network, hence they produce a significant part of the data used for training. This can explain why the models are more precise when they predict speed for this group.

We noticed a huge prediction error for metropolitan bus line 532 with resp RMSE, MAE and MAPE at 12.1, 11.9 and 427%. We investigated how can the model perform that bad on some cases and found

that the average speed of the only unknown inter-station of the bus line 532 is lower than 5km/h, while its features are alike others inter-stations in the network. The speed calculation business rules used might yield such a low speed, e.g if the bus has to wait at a bus stop for operational reasons. In other words, inter-stations that have very low average speed with non specific features set is prone to yield important predictions errors (i.e considered as an outlier).

Globally, micro-learning yields variable quality results, with a higher error rate for unknown inter-stations. The features used for predictions probably need enhancement in terms of noise reduction and/or external data addition such as smart-card data, traffic status, etc.

4.3 Discussion

Micro-learning on graphs at edge level (e.g. bus-inter-stations level) yields variable results on a real data set. Prediction error tends to be more important when no observations on this very edge is available in the learning data set. These results point the importance of having high quality data and choosing the right features set for the model. However, Figure 3 shows that the micro learning model has a sensitivity to current features, hence this allows for prospective scenarios, when one needs to test different configurations for a future bus line.

Comparing compositional prediction results (Compred) with Macro prediction, raised the following remarks: For urban group, Compred MAPE and RMSE differences are only of 0.5% and 0.2 km/h while Compred's MAE is 0.2 km/h better than macro's. That is to say, macro and Compred are nearly on par when it comes to predict urban bus speed. In other groups, compositional prediction performs slightly better than macro predictions, with MAE, RMSE and MAPE lower for Compred than macro by around 2.5 to +25% depending on the measure we compare. Hence, this suggests that compositional prediction seems to be more capable of catching fine grain variations (i.e variation on inter-stations) than macro does, as shown in Table 3. This table shows the average predicted speed variation (standard deviation in km/h) of bus lines over time (holidays, days, period). We observed that Compred model seems to be at least twice as sensitive to time features as macro model.

We compared macro-learning (i.e. building models based only on the measures of entire paths) with compositional prediction based on micro-learning (full knowledge of all measures on edges). Macro-learning is disadvantaged by a partial knowledge of the network, while Compred is disadvantaged by a higher volume of data and its obligation to perform an aggregation of small-scale predictions, which is mandatory in prospective scenarios. It appears that the obtained quality is good. Actually our method reaches state of the art performance if we compare with [4]. Finally, the micro-learning mixed with compositional approaches allow the comparison of different configurations at any scale within the bus line.

4.4 Threats to validity

As internal threats, we acquired the real data set ourselves using our own code, which may be prone to errors. But these data are used in production by the Keolis company, and has been controlled using business rules many times. Also, we used only complete data. No imputation was done. Our code uses a high-level language and state-of-the art libraries for data extraction and machine learning. As construct validity, we choose to use Random forests as ensemble methods model for regression. The choice we made was based on performance amongst a set of models tested on a small sample of data from which random forests performed better (in terms of results and computing time). However, we assumed that the models would behave in an analogous manner with a wider dataset, whereas there is a small risk that this is not the case. Finally, the scope of this paper is to assess the quality of compositional prediction for composite objects in complex environment against traditional macro approaches, we then considered the machine-learning model selection as a secondary issue, hence we probably could have different results with other models. Yet, choosing better models for compositional prediction is another issue that probably needs a dedicated work. As an external threat, our data set, time frame and targets selection might have specificities that

could prevent generalization beyond these lines or Rennes Metropolis. On the other side, we gathered 15M tuples over a year from a large city common bus transportation system. The chosen lines are typical and of different kind (urban, inter-district, metropolitan and express).

5 Related Work

5.1 Micro learning

Hartmann et. al. developed Greycat[20,22], a tool that provides a dynamic temporal graph approach for fast evolving networks. Our concept is different because our goal is to use the history of the network to build a new path in the network using the knowledge we got from the current network.

5.2 Link weight prediction

Hou et. al [12] developed Model R a deep neural network model that aims to predict the existence and the weight of new edges (links) within a graph. Our model comes as support for the prediction of weights of both existing and new edges for which features are known, given a timestamp. Also the machine learning model used for the predicting could easily be changed.

Kumar et. al [13] proposed an algorithm to predict edges weight in Weighted Signed Networks (WSN). They predict weight using two vertex based metrics named fairness and goodness, hence their method is vertex centric. A contrario, ours focuses on edges features and aims at predicting variables edges weight given different sets of features.

Zhao et. al [26] proposed a way of predicting the existence of edges and their weight using reliable route approach. They use local similarity measures which consists of examining vertex neighbourhood to determine a reliable route, hence edges and weights. Their work relies on the intrinsic properties of graphs while ours uses graph as a support for real world data representation and exploration. Indeed, while the neighbourhood of nodes is used to predict edges and weight, we only rely on real world features (OSM and historical data) without using any of the graph theory properties.

Fu et. al [11] proposed link weight prediction combining original graph and line graph properties. They also use the graphs intrinsic properties such as degrees to predict link weight over evolving networks. While their work focuses on graph structure to generate features, we gather ours from domain knowledge. Also our method is edge focused and straight forward: We try to keep the network as it is in real life which does not need graph conversion to line graph.

B. Taskar et. al. [21] studied the predictability of links within a relational graph. As an example they used neural networks to try to predict relationships between people on social network data. Their work differ from ours to the extent that they try to predict the existence of new edges while we create new edges knowing their properties and use them to predict their weight.

5.3 Bus travel time / bus speed prediction

Mendes-Moreira and Barachi [16] imagined a prediction model for networks by predicting sub parts of the networks and re-conciliate the aggregated predictions of the sub-parts with the path they are part of. They do this using a method they called Reconciliation For Regression (R4R) by weighing every sub-predictions using a constraint least square algorithm. Their results show that they reach state of the art performance for bus travel time prediction. However, it is unclear on how far from the reality their model perform without MAPE. One could also raise the following statement: the added complexity of R4R is questionable because it shows that it seems to never offer a better improvement than 3% in prediction precision when compared to other models, including simple ones such as Multivariate Linear Regression (MLR).

Fernandez et. al [10] proposed a statistical model to predict bus commercial speed. Their model needs a lot of calibration made by hand while ours just need data collection.

Petersen et. al [17] predict travel time on known bus lines in order to enhance the quality of service. We try to predict a commercial speed for a not yet existing bus line for strategic purpose, hence giving hints on how a new bus line would likely behave.

A recent work [2] considered prediction in a road network, by decomposing the learning task down to the road segment. They differ with us as their consider a route prediction task (what is the next segment), and do not consider non-existing routes as we do.

Another recent work [25] propose a mathematical model to predict bus time of arrival at bus stop using surrounding traffic and signal control. Their model do it in real time and could be used to enhance travelers information. However their work do not cover the exact same areas as ours.

M. Altinkaya and M. Zontul[1] worked on a review of computational models for bus arrival time predictions. They analyze different methods to predict bus arrival time including statistical and machine learning ones. They suggest that it is needed to focus on data that look alike in order to diminish the prediction error. Which is what we try to do using micro learning and time periods selection.

5.4 Traffic simulation

Barcelo et al. [3] proposed a software based on AIMSUN NG, a traffic simulation tool, that integrates MACRO, MESO & MICRO simulation in a single framework, sharing a unique database. Doing so allows the solving of the consistency issues when shifting between MACRO, MESO and MICRO scales thanks to data aggregation rules. Their work differ from ours as long as their tools aim is to enhance the capabilities of simulation tools.

Burghout et al. [5] worked on a hybrid mesoscopic-microscopic traffic simulation that lets one apply microscopic simulation on area of interest in mesoscopic areas. They claim that mesoscopic simulation is not as needy as microscopic simulation. We confirm this claim but using a machine learning approach.

Mendes-Moreira and Barachi [16] imagined a prediction model for networks by predicting sub parts of the networks and re-conciliate the aggregated predictions of the sub-parts with the path they are part of. They do this using a method they called Reconciliation For Regression (R4R) by weighing every sub-predictions using a constraint least square algorithm. Their results show that they reach state of the art performance for bus travel time prediction. However, it is unclear on how far from the reality their model perform without MAPE. One could also raise the following statement: the added complexity of R4R is questionable because it shows that it seems to never offer a better improvement than 3% in prediction precision when compared to other models, including simple ones such as Multivariate Linear Regression (MLR).

6 Conclusion and Future work

In this work, we presented compositional prediction on graphs, an approach to infer path properties from edge properties obtained by micro-learning. Based on a real-size application, we evaluated this approach with respect to classical macro-learning. We showed that it allows to compare different configurations for a future bus line, at variable scale in the bus line, enabling the study of prospective scenarios. This approach exhibits at least comparable and often better quality than the classical approach, while offering state of the art performance with prospective capabilities.

As a future work, we would like to finely understand the quality gap between macro-learning and compositional prediction: while good on small scales, it could be reasonable to switch to macro-prediction on relevant sub-path of the graphs. Also our model must be tested on other scenarios that become crucial for bus network operators such as the fuel consumption along a bus line, or with other composition methods such as product e.g.; computing the risk of accident along a path given its sub-parts individual risks. Eventually

given the overestimation of the model one would try to apply arbitrated ensemble methods [6] to re-qualify the model output and enhance its precision. Finally, since the presented model could be embedded within a data exploration model, this paper can be seen as a first step toward declarative languages for prospective scenarios.

References

1. Altinkaya, M., Zontul, M.: Urban bus arrival time prediction: A review of computational models. IJRTE (2013)
2. Amirat, H., Lagraa, N., Fournier-Viger, P., Ouinten, Y.: Myroute: A graph-dependency based model for real-time route prediction. JCM **12**, 668 (2017)
3. Barceló, J., Casas, J., García, D., Perarnau, J.: Methodological Notes on Combining Macro, Meso and Micro Simulation Models for Transportation Analysis. In: Workshop on Modeling and Simulation. Sedona, AZ (2005)
4. Berger-Wolf, T., Chawla, N. (eds.): Proceedings of the 2019 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, Philadelphia, PA (May 2019). https://doi.org/10.1137/1.9781611975673
5. Burghout, W., Koutsopoulos, H., Andréasson, I.: Hybrid Mesoscopic-Microscopic Traffic Simulation. Transportation Research Record: Journal of the Transportation Research Board **1934**, 218-25 (2005)
6. Cerqueira, V., Torgo, L., Pinto, F., Soares, C.: Arbitrated ensemble for time series forecasting. In: Ceci, M., Hollmén, J., Todorovski, L., Vens, C., Džeroski, S. (eds.) Machine Learning and Knowledge Discovery in Databases. pp. 478–494. Springer, Cham (2017)
7. Cortés, C.E., Gibson, J., Gschwender, A., Munizaga, M., Zúñiga, M.: Commercial bus speed diagnosis based on GPS-monitored data. Transportation Research Part C: Emerging Technologies **19**(4), 695–707 (8 2011). https://doi.org/10.1016/j.trc.2010.12.008
8. Courtois, X., Dobruszkes, F.: L'(in)efficacité des trams et bus à Bruxelles, une analyse désagrégée. Brussels Studies. La revue scientifique électronique pour les recherches sur Bruxelles / Het elektronisch wetenschappelijk tijdschrift voor onderzoek over Brussel / The e-journal for academic research on Brussels (6 2008). https://doi.org/10.4000/brussels.603
9. Fernandez, R., Valenzuela, E.: A model to predict bus commercial speed. Traffic Engineering & Control **44**(2) (2 2003)
10. Fernandez, R., Valenzuela., E.: A model to predict bus commercial speed. Traffic Engineering & Control 44 **2** (2003)
11. Fu, C., Zhao, M., Fan, L., Chen, X., Chen, J., Wu, Z., Xia, Y., Xuan, Q.: Link Weight Prediction Using Supervised Learning Methods and Its Application to Yelp Layered Network. IEEE Transactions on Knowledge and Data Engineering **30**(8), 1507–1518 (Aug 2018). https://doi.org/10.1109/TKDE.2018.2801854
12. Hou, Y., Holder, L.B.: On Graph Mining With Deep Learning: Introducing Model R for Link Weight Prediction. Journal of Artificial Intelligence and Soft Computing Research **9**(1), 21–40 (Jan 2019). https://doi.org/10.2478/jaiscr-2018-0022
13. Kumar, S., Spezzano, F., Subrahmanian, V.S., Faloutsos, C.: Edge Weight Prediction in Weighted Signed Networks. In: 2016 IEEE 16th International Conference on Data Mining (ICDM). pp. 221–230. IEEE, Barcelona, Spain (Dec 2016). https://doi.org/10.1109/ICDM.2016.0033
14. Ma, X., Chen, X.: Public transportation big data mining and analysis. In: in Data-Driven Solutions to Transportation Problems. Elsevier (2019)
15. Matsumoto, T., Sakakibara, K., Tamaki, H.: Bus line optimization using multi-agent simulation model of urban traffic behavior of inhabitants applying branch and bound techniques. pp. 234–239. IEEE (Jul 2015). https://doi.org/10.1109/SICE.2015.7285551

16. Mendes-Moreira, J., Baratchi, M.: Reconciling Predictions in the Regression Setting: An Application to Bus Travel Time Prediction. In: Berthold, M.R., Feelders, A., Krempl, G. (eds.) Advances in Intelligent Data Analysis XVIII, vol. 12080, pp. 313–325. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-44584-3_25, series Title: Lecture Notes in Computer Science
17. Petersen, N.C., Rodrigues, F., Pereira, F.C.: Multi-output bus travel time prediction with convolutional lstm neural network. Expert Systems with Applications **120**, 426–435 (2019)
18. Pontius, R.G., Thontteh, O., Chen, H.: Components of information for multiple resolution comparison between maps that share a real variable. Environmental and Ecological Statistics **15**(2), 111–142 (Jun 2008). https://doi.org/10.1007/s10651-007-0043-y
19. Robinson, S., Narayanan, B., Toh, N., Pereira, F.: Methods for pre-processing smartcard data to improve data quality. Transportation Research Part C: Emerging Technologies **49**, 43–58 (dec 2014). https://doi.org/10.1016/j.trc.2014.10.006
20. T. Hartmann, A. Moawad, F.F., Traon, Y.L.: The next evolution of mde: A seamless integration of machine learning into domain modeling. SoSyM (2017)
21. Taskar, B., Wong, M.F., Abbeel, P., Koller, D.: Link prediction in relational data. Advances in neural information processing systems **16**, 659–666 (2003)
22. Thomas, H., Fouquet, F., Moawad, A., Rouvoy, R., Traon, Y.L.: GreyCat: Efficient What-If Analytics for Data in Motion at Scale. IS **83**, 101–117 (2019)
23. Treethidtaphat, Wichai, W.P.A.e.S.K.: Bus arrival time prediction at any distance of bus route using deep neural network model. International Conference On Intelligent Transportation (2017)
24. Zaki, M., Ashour, I., Zorkany, M., Hesham, B.: Online Bus Arrival Time Prediction Using Hybrid Neural Network and Kalman Filter Techniques. IJMER **3**, 2035–2041 (2013)
25. Zhang, H., Liang, S., Han, Y., Ma, M., Leng, R.: A prediction model for bus arrival time at bus stop considering signal control and surrounding traffic flow. IEEE Access **8**, 127672–127681 (2020)
26. Zhao, J., Miao, L., Yang, J., Fang, H., Zhang, Q.M., Nie, M., Holme, P., Zhou, T.: Prediction of Links and Weights in Networks by Reliable Routes. Scientific Reports **5**(1), 12261 (Dec 2015). https://doi.org/10.1038/srep12261