Online Collective Demand Forecasting for Bike Sharing Services

Charles Dickens
University of California Santa Cruz
cadicken@ucsc.edu

Alexander Miller
University of California Santa Cruz
almumill@ucsc.edu

Lise Getoor
University of California Santa Cruz
getoor@ucsc.edu

Abstract

We introduce a general time-series forecasting method that extends classical seasonal autoregressive models to incorporate exogenous and relational information in an online setting. Our approach is implemented using the probabilistic programming language Probabilistic Soft Logic (PSL) [2]. We leverage recent work that enables the scalable application of PSL to online problems and propose novel modeling patterns to leverage dependencies between multiple time series. We demonstrate the applicability and performance of our method for the task of station level demand forecasting on three bike sharing systems. We perform an analysis of the demand time series and present evidence of relational dependencies among the stations, motivating the need for a forecasting model that leverages the rich relational structure in the bike sharing networks. Our approach significantly improves multi-step forecasting accuracy of autoregressive time-series models on all three datasets. Further, our approach is easily extendable and we expect applicable to a variety of other time-series forecasting problems.

Keywords: time-series, demand forecasting, bike-sharing

1. Introduction

Shared mobility services are a cost effective and sustainable method for supporting the adoption of public transportation, thereby reducing traffic congestion and carbon emissions [8]. A shared mobility service acts as a solution for last mile transportation, conveniently connecting users from bus and train stops to their final destination [18]. A dock or station-based mobility service hosts multiple stations positioned throughout an urban area where users can check out and return a bike or scooter. Examples of this type of shared mobility service can be found in many major cities.

Adoption and retention of shared mobility services relies heavily on their dependability. A central problem for the service operators is continuously *rebalancing* the amount of vehicles at each station [18]. Rebalancing is crucial since a station with no open docks forces a user to return a vehicle to a station that is not on their intended route. Analogously, having too few vehicles limits the number of users that can depart from a station. Simply monitoring the station status, i.e., whether a station is full or low on vehicles will lead to imbalances and not allow for strategic planning to satisfy demand. Thus, predictive analytics plays an important role in the rebalancing problem.

Accurate forecasts of station-level arrival and demand rates support the application of one of the many existing algorithms for solving the rebalancing problem. However, scalable and accurate station-level forecasts are difficult to make. Each of the many individual stations has its own complex demand pattern. For instance, a station positioned at a busy train stop will have a very different demand time-series than one in a residential district, and both may have long-term seasonal dependencies. Furthermore, exogenouos factors and dependencies between stations in the network can have a significant impact on the demand. An exogenous factor such as abnormally hot or cold weather will influence the demand on a given day. Moreover, the demand and arrival rates for two stations on a common route will be highly correlated. We aim to fill this need in the shared mobility service literature with a highly scalable and expressive framework for providing station-level demand forecasts.

In this work, we introduce a novel framework that extends traditional autoregressive models to capture a variety of structural dependencies. Our approach makes use of soft probabilistic dependencies, specified using Probabilistic Soft Logic (PSL) [2], an expressive and highly scalable probabilistic programming language. The models we develop are interpretable and extendable to a variety of temporal demand prediction tasks. We show how to extend existing PSL work for online inference to support the updates required to reproduce classical autoregressive forecasts. We then extend the model to reason with complex dependencies between time series, long-term temporal seasonality, and exogenous factors to collectively infer future demand. We show that introducing this collective reasoning achieves scalable and statistically-significant improvements in both mean and median R^2 and RMSE performance for station level forecasts on all datasets. Moreover, we find that, surprisingly, an alternative multi-step forecasting technique our proposed framework supports further improves performance over classical autoregressive models.

2. Related Work

Improving the reliability of shared mobility services has a positive impact on user experience and participation, as shown by both Yu et al. (2018) and Kabra et al. (2018). For this reason, operators preemptively rebalance the distribution of the service's bicycles to ensure users have access when and where they need it. Laporte et al. (2018) provides an excellent description of rebalancing and other operations problems that arise in shared mobility services and surveys techniques for solving them. Notably, the operations research community has contributed effective formulations of rebalancing objectives and scalable techniques for optimization [4, 23, 27, 12, 3]. In this work, we focus on improving the quality of demand forecasts that can in turn be used to support rebalancing solutions.

Demand forecasting for bike-sharing services is an active area of research and many modern machine learning-based techniques, surveyed by Albuquerque et al. (2021), have been proposed for the task. Initial work on demand forecasting in shared mobility services developed independent time-series forecasts for each station depending purely on historical demand observations. For instance, Schuijbroek et al. (2017) forecasts bike availability using independent and specialized Markov chain models. Schuijbroek et al. (2017) fit the Markov chain models using historical demand time series and apply the forecasts to formulate a mixed-integer program to design a rebalancing schedule. Similarly, Cagliero et al. (2017) introduced station specialized Bayesian and associative models to forecast critical events at stations in a bicycle sharing service.

Recent research from Eren and Emre UZ (2020), Kon et al. (2021), and Li et al. (2022) showed external factors such as weather, geographic topology, and other public transportation systems have a substantial affect on a bike station's demand. Thus improvements in demand forecasting accuracy have been made by considering exogenous variables. For instance, Hulot et al. (2018) and Ruffieux et al. (2018) advanced station level demand forecasting by training specialized machine learning models for individual stations that make use of temperature, humidity, and other weather-based features. Moreover, Hulot et al. (2018) overcame overfitting with dimensionality reduction techniques, some of which leverage the observation that stations can be organized into clusters with similar demand patterns. Another line of work, exemplified by Xu et al. (2018), applies deep learning techniques to model demand time-series with exogenous effects. Xu et al. (2018) use a long short-term memory (LSTM) model that takes historical observations and weather data to forecast demand in the station-free setting. Motivated by the success of these methods, the model we introduce is designed to consider multiple exogenous variables to make forecasts in the station-based setting.

Beyond the historical demands and exogenous information, shared mobility services have many structural dependencies that provide a signal for the future demand. For example, Rudloff and Lackner (2014) identified if a nearby station is empty then its demand will overflow to a nearby station. This supply and demand dependency, along with an exogenous variable, is applied to improve the precision of distributions modelling the departure and arrival rates at a station. Likewise, Faghih-Imani and Eluru (2016) verified that nearby stations in the New York CitiBike bicycle sharing service consistently have correlated demand rates. The authors incorporated the dependencies with spatial lag and error models. Work by Lin et al. (2018) also aims to exploit structural dependencies in the same CitiBike service network. The authors propose the use of graph convolutional networks (GCN) to learn pairwise correlations between bike stations and techniques for defining the graph structure of the model. Specifically, spatial distance between stations, route frequency, route duration, and demand correlation are used to define an adjacency matrix for the GCN. Our proposed approach contributes to this line of research with a framework that leverages complex dependencies between stations, exogenous effects, and historical observations.

3. Demand Forecasting in Bike Share Services

We analyze the demand patterns and structural properties of three active bike sharing services in the San Francisco Bay Area [22], Boston [5], and Los Angeles [21]. The bike share datasets follow a similar data format, each containing both station information (coordinates of the station and the number of docks) and trip information (start and end station and times of each trip). Table 1 shows the number of stations and the minimum start and maximum end dates of the recorded trips.

Dataset	Number of Stations	Start Date	End Date
Bay Area	68	8/29/2013	5/1/2015
Boston	187	4/3/2013	4/3/2014
Los Angeles	272	2/1/2018	8/1/2019

Table 1: The number of stations and the start and end dates of the trip data available in each of the three datasets.

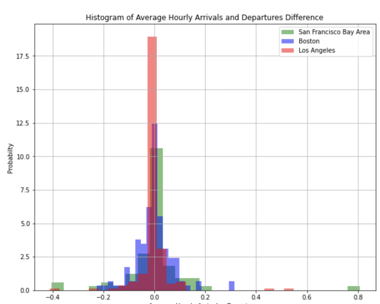

Figure 1: A normalized histogram plotting the average difference of hourly arrivals and departures at stations for three bike share services. Many stations are, on average, imbalanced.

Users experience a service failure when a station is full and they cannot return a bike, or a station is empty and they cannot check out a bike. Imbalance of available bikes at a station occurs when the rate of arrivals and departures of trips is unequal. We define the *departure demand* and *arrival demand* at a station for a given hour to be the total number of trips starting and ending at the station in the hour if bikes and station docks were not limited by availability. Observed departure and arrival rates are used as proxy measurements of demand for bikes and open docks, respectively. Figure 1 illustrates the level of demand imbalance occurring at individual stations in the three bike share services. In a self-balancing network, the difference between hourly arrival and departure rates would be zero for every station in the network. However, for many stations in the networks, this is not the case.

To address the issue of demand imbalance, operators regularly need to redistribute bikes in anticipation of demand. Strategic rebalancing requires a forecast of both the arrival and the departure demands for individual stations. Figure 2 shows the demand time series of a selected station from each of the three bike share datasets. The results of a KPSS test indicate that the majority of demand time-series are stationary [17]. Furthermore, there is significant autocorrelations at 1 and 2 hour lags and seasonal autocorrelations at 12, 24, and 168 hour lags. This analysis indicates that a seasonal autoregressive model with no differencing at the mentioned lags is an appropriate time-series model for station demand.

Dataset	Random	Nearby	Commute	Cluster
Bay Area	0.193	0.328	0.394	0.376
Boston	0.330	0.498	0.451	0.512
Los Angeles	0.036	0.120	0.186	0.068

Table 2: Average time series correlation coefficient for pairs of stations with different relations.

,

However, there is additional structure in bike share networks that cannot be effectively leveraged by independent autoregressive time-series models. Table 2 shows the average correlation coefficients for pairs of stations with different types of relations. For instance, correlation between the demand time series of pairs of stations within a 0.5 KM radius (Nearby) is considerably higher than the average correlation between two randomly selected stations in the network (Random). Furthermore, the departure demand for a station is correlated with the arrival demand for its most common destinations (Commute). In other words, if many trips starting from one station have been observed to end at a specific station, i.e. they have been identified as a commute pair, then we observe a correlation between the two stations' departure and arrival demands, respectively. Further, applying hierarchical agglomerative clustering to stations based on historical demand patterns (Cluster) results in groups of above average correlated station pairs.

4. Online Probabilistic Soft Logic

Inspired by the structure revealed in the data analysis, we introduce a model for jointly reasoning with related time series. Our model is built using

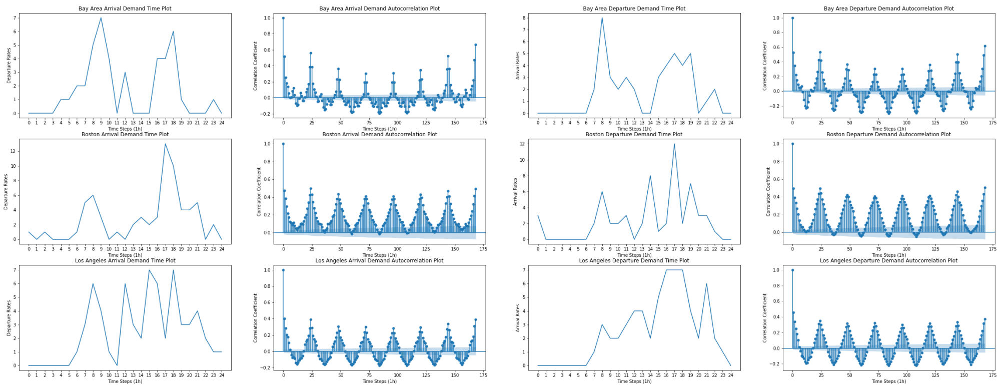

Figure 2: Arrival and departure demand time series and autocorrelation plots of the departure rates recorded at selected stations in the San Francisco Bay Area, Boston, and Los Angeles bike share services.

an expressive and scalable probabilistic programming language called Probabilistic Soft Logic (PSL) [2].

4.1. Hinge-Loss Markov Random Fields

PSL instantiates a tractable class of graphical models called hinge-loss Markov random fields (HL-MRF). A collection of arithmetic and first-order logical statements, referred to as *rules*, determine dependencies between *atoms*: the attributes and relations present in a domain. For instance, in PSL we represent the value of a time series at at time T with the atom $\textsc{Series}(\texttt{T})$. Then, multiple factors that have an influence on our forecast of the time series are expressed as rules. We capture exogenous factors, such as weather, with the atom $\textsc{Exog}(\texttt{T})$. Moreover, an existing time series predictor is modeled with the atom $\textsc{Series}(\texttt{T})$. The following are examples of a weighted logical and arithmetic rule relating the atoms.

$$w_1 : \textsc{Exog}(\texttt{T}) \rightarrow \textsc{Series}(\texttt{T}) \tag{1}$$

$$w_2 : \textsc{Predictor}(\texttt{T}) = \textsc{Series}(\texttt{T}) \tag{2}$$

The first rule forces values of the series to be greater than or equal to the value of the exogenous effect at the same time $\texttt{T}$. The second rule forces the value of $\textsc{Series}(\texttt{T})$ to be equal to the value of $\textsc{Predictor}(\texttt{T})$. Sometimes the two rules cannot both be satisfied at the same time. The *weight* of a PSL rule, denoted by the variables w_1, w_2 in (1) and (2), is a non-negative value representing the importance of satisfying the rule in the model. The higher the relative value of a weight, the more important the corresponding rule is to satisfy.

Weighted rules in PSL are not strict constraints. Rather, each weighted rule is a template for instantiating soft constraints, or hinge-loss *potentials*. The variables in the atom arguments are replaced by constants in the data to create *ground rules*. For instance, consider a dataset containing entries for the atoms referenced in (1) and (2) for times $T \in \{t_1, t_2\}$. The following ground rules will be instantiated for the the rule (1):

$$w : \textsc{Exog}(t_1) \rightarrow \textsc{Series}(t_1)$$
$$w : \textsc{Exog}(t_2) \rightarrow \textsc{Series}(t_2)$$

Ground rules are translated into potentials using Łukasiewicz logic. The potentials are defined using variables which map to the unique instantiations of the atoms. Specifically, the vectors of all unobserved and observed variables are denoted by $\mathbf{y} = [y_i]_{i=1}^{n}$ and $\mathbf{x} = [x_i]_{i=1}^{n'}$, respectively. In PSL, atoms and their corresponding variables, take on continuous values in the closed interval $[0, 1]$. All potentials in an HL-MRF have the following functional form:

$$\phi(\mathbf{y}, \mathbf{x}) = (\max\{\ell(\mathbf{y}, \mathbf{x}), 0\})^q \tag{3}$$

where $\ell(\mathbf{y}, \mathbf{x})$ is a linear function of the variables $\mathbf{y}$ and $\mathbf{x}$, and the exponential term, $q \in \{1, 2\}$, is a hyperparameter set by the modeler.

The set of all m potentials, $(\phi_1, \cdots, \phi_m)$ created during grounding with corresponding weights $\mathbf{w} = (w_1, \cdots, w_m)$ are used to define the HL-MRF energy function.

$$f(\mathbf{y}, \mathbf{x}, \mathbf{w}) = \sum_{i=1}^{m} w_i \phi_i(\mathbf{y}, \mathbf{x}) \tag{4}$$

The energy function then defines the HL-MRF conditional distribution.

$$P(\mathbf{y}|\mathbf{x}; \mathbf{w}) = \frac{1}{Z(\mathbf{w}, \mathbf{x})} \exp(-f(\mathbf{y}, \mathbf{x}, \mathbf{w})) \tag{5}$$

where $Z(\mathbf{w}, \mathbf{x}) = \int_{\mathbf{y}} \exp(-f(\mathbf{y}, \mathbf{x}, \mathbf{w}))d\mathbf{y}$ is the partition function.

Maximum a posteriori (MAP) estimation is used to obtain a joint prediction of the unobserved variables.

$$\arg\max_{\mathbf{y}\in[0,1]^n} P(\mathbf{y}|\mathbf{x};\mathbf{w}) = \arg\max_{\mathbf{y}\in[0,1]^n} \sum_{i=1}^{m} w_i \phi_i(\mathbf{y}, \mathbf{x}) \quad (6)$$

A key advantage of HL-MRFs is that MAP inference is a tractable convex problem [2].

4.2. Online Collective Inference

Recent work enabled the application of PSL in online settings where new observed and unobserved atoms are revealed sequentially [9, 24]. For HL-MRFs, the task of maintaining and performing inference over a sequence of evolving graphical models is referred to as *online collective inference*, shown in Algorithm 1.

Algorithm 1: Online Collective Inference

Data: $\mathbf{x}_0, \mathbf{y}_0, \mathbf{w}, \mathcal{T}$

1 **for** $t = 0, 1, 2, \cdots$ **do**

2   Ground $P(\mathbf{y}_t|\mathbf{x}_t;\mathbf{w})$

3   Predict $\mathbf{y}_t^* \in \arg\max_{\mathbf{y}\in[0,1]^{n_t}} P(\mathbf{y}|\mathbf{x}_t;\mathbf{w})$

4   Receive model updates to obtain $\mathbf{x}_{t+1}, \mathbf{y}_{t+1}$

5 **end**

The evolution of the graphical model is encoded by sequences of *model updates*. In general, a model update can be classified into one of the following three categories: value update of an observed variable $x_i \in X$, addition or deletion of a random variable $x_i \in X$ or $y_i \in Y$, or the addition or deletion of a rule. We develop the online PSL framework by extending the set of supported model updates. Specifically, we implemented a command, $FixAtom$, for moving a variable from the unobserved set Y to the observed set X with its inferred MAP value. This command circumvents the expensive process of first deleting variables in $y_i \in Y$, then adding to $x_i \in X$ with inferred values, and re-grounding the rules. The $FixAtom$ command makes preforming multi-step forecasts highly efficient. Details on the process of obtaining multi-step forecasts are provided in the following section.

5. Collective Time-Series Forecasting

We propose a novel framework for performing collective time-series forecasting using HL-MRFs. We present our approach in the context of forecasting the arrival and departure rates at stations in a bike sharing network. Thus, the two target atoms in the PSL model are $\textsc{Arrive}(\mathtt{S}, \mathtt{T})$ and $\textsc{Depart}(\mathtt{S}, \mathtt{T})$, which represent the normalized count of arrivals and departures, respectively, for a station $\mathtt{S}$ during a period of time $\mathtt{T}$. We organize the presentation of the PSL rules into four subsections to highlight their modeling contribution: 1) seasonal autoregression, 2) exogenous factors, 3) collective reasoning, 4) local prediction, and 5) regularizers. We emphasize that the rules presented here are also applicable as general modeling patterns for time-series forecasting problems.

5.1. Seasonal Autoregression

Classical autoregressive (AR) models represent time-series variables as functions of its lagged values [6, 14]. Seasonal-AR (S-AR) models are an AR extension that additionally regress onto seasonal lags of the time-series. Formally, given time-series data $\mathbf{y} = [y_t]_{t=1}^{T}$, an S-AR model with AR order p and seasonal order P at a period s is given by:

$$y_t = \sum_{i=1}^{p} \alpha_i y_{t-i} + \sum_{i=1}^{P} \beta_i y_{t-i\cdot s} + \gamma + \epsilon_t \qquad (7)$$

where $\alpha = (\alpha_1, \cdots, \alpha_p)$, $\beta = (\beta_1, \cdots, \beta_P)$, and γ are the parameters of the model and ϵ_t is a mean 0 error term. One prevalent approach to fitting the model parameters is by minimizing the total squared error of the process.

$$\min_{\alpha,\beta,\gamma} \sum_{t=1}^{T} (y_t - (\alpha_1 y_{t-1} + \cdots + \alpha_p y_{t-p} + \beta_1 y_{t-s} + \cdots + \beta_P y_{t-P\cdot s} + \gamma))^2 \qquad (8)$$

With parameters fit on an observed portion of a time-series, multi-step forecasts are made incrementally. Precisely, an h step forecast begins with computing y_{T+1} using (7). Then, the forecasted value of y_{T+1} is used to obtain y_{T+2}, and so on, until all h steps are computed.

The described S-AR model for a normalized time-series can be implemented and extended with online PSL. The rules for expressing auto-regressive relations take the following form:

$$
\begin{aligned}
w_{k_1} &: \text{DEPART}(\texttt{S}, \texttt{T1}) \\
&\quad = \text{DEPART}(\texttt{S}, \texttt{T2})\{\text{LAG}_k(\texttt{T1}, \texttt{T2})\} && (9) \\
w_{k_2} &: -\text{DEPART}(\texttt{S}, \texttt{T1}) \\
&\quad = \text{DEPART}(\texttt{S}, \texttt{T2})\{\text{LAG}_k(\texttt{T1}, \texttt{T2})\} && (10) \\
w_{k_3} &: \text{DEPART}(\texttt{S}, \texttt{T}) && (11) \\
w_{k_4} &: !\text{DEPART}(\texttt{S}, \texttt{T}) && (12)
\end{aligned}
$$

The $\text{LAG}_k(\texttt{T1}, \texttt{T2})$ atom is binary valued and represents whether the time T1 is a fixed lag k of the time T2. Only the groundings of the rule such that $\text{LAG}_k(\texttt{T1}, \texttt{T2})$ is true will be instantiated. Then, for the triple of arguments $(s, t, t - k)$ such that $\text{LAG}(\texttt{t - k}, \texttt{t})_k = 1.0$ let $y_t = \text{DEPART}(\texttt{s}, \texttt{t})$ and $y_{t-k} = \text{DEPART}(\texttt{s}, \texttt{t - k})$. The following potentials are instantiated from the rules for the variables y_t and y_{t-k}:

$$\phi_{k_1}(y_t, y_{t-k}) = (y_t - y_{t-k})^{q_1} \tag{13}$$

$$\phi_{k_2}(y_t, y_{t-k}) = (y_t + y_{t-k})^{q_2} \tag{14}$$

$$\phi_{k_3}(y_t, y_{t-k}) = (1 - y_t)^{q_3} \tag{15}$$

$$\phi_{k_4}(y_t, y_{t-k}) = y_t^{q_4} \tag{16}$$

Therefore, a PSL model of purely autoregressive rules with $q_1 = q_2 = 2$ and $q_3 = q_4 = 1$ and a single target variable y_t instantiates the HL-MRF energy function:

$$
\begin{aligned}
f(y_t) = \sum_i & \left(w_{k_{i,1}}(y_t - y_{t-k_i})^2 + w_{k_{i,2}}(y_t + y_{t-k_i})^2\right) \\
& + w_{k_3}(1 - y_t) + w_{k_4} y_t
\end{aligned} \tag{17}
$$

Then, first order optimality conditions for this convex function with weights $\mathbf{W} \in \Delta^r$ implies that

$$
\begin{aligned}
&\arg\max_{y_t} f(y_t) \\
&= \left(\sum_i (w_{k_{i,1}} - w_{k_{i,2}}) y_{t-k_i}\right) + \frac{w_{k_4} - w_{k_3}}{2}
\end{aligned} \tag{18}
$$

Thus, the MAP state of the HL-MRF has a closed form solution that is a linear combination of its lags. Further, observe that although weights are restricted to non-negative values to preserve the convexity of inference, negative correlations and constants can still be expressed.

The $FixAtom$ command is used to perform multi-step predictions via incremental forecasting. Precisely, an h step incremental forecast in online PSL begins with adding a single target atom y_{T+1} and setting it to the MAP state of the HL-MRF. Next, the target atom y_{T+1} is fixed to its current value and the next target atom y_{T+2} is added. This process is repeated until all h steps are computed.

Alternatively, the multi-step forecast can be computed all at once. This is achieved by adding all the target atoms $y_{T+1}, \cdots, y_{T+h}$ and finding the joint MAP state of the HL-MRF. We refer to this method as *batch* forecasting. Note that batch forecasting does not have a closed form solution as does the incremental approach.

5.2. Exogenous Factors

Exogenous factors are external variables that influence the time-series. For instance, in the bikeshare setting, a rainy day can have a negative effect on the demand level, and hence decrease both the arrival and departure rates. Conversely, busy commute hours on weekdays typically see higher demand. The PSL rules expressing these relations for departure rates are shown below:

$$w : \text{RAINING}(\texttt{S}, \texttt{T}) \rightarrow \neg\text{DEPART}(\texttt{S}, \texttt{T}) \tag{19}$$

$$w : \text{COMMUTEHOUR}(\texttt{S}, \texttt{T}) \rightarrow \text{DEPART}(\texttt{S}, \texttt{T}) \tag{20}$$

In these rules, RAINING and $\text{COMMUTEHOUR}(\texttt{S}, \texttt{T})$ are binary valued atoms reflecting whether or not there is rainy weather at station S at a time T and whether time T is a commute hour, respectively.

5.3. Collective Forecasting

Collective predictions are made jointly, rather than independently, allowing the model to capitalize on the relational information and structure present in the data. Atoms that are connected by rules appear together in HL-MRF potential functions. Then, a MAP state of the HL-MRF density function is found by jointly optimizing over the unobserved variables to minimize the total sum of the potentials. In other words, a state of the random variables is found that most satisfies the rules.

For instance, spatial relations and data-driven techniques can cluster stations with positively correlated demands. For example, geographically nearby stations tend to have similar patterns of utilization. Moreover, employing a clustering algorithm (e.g., hierarchical agglomerative clustering) provides clusters of stations with correlated historical demand time series, which also can be utilized. Let the atom $\text{ADJ}(\texttt{S1}, \texttt{S2})$ and $\text{CLUSTER}(\texttt{S1}, \texttt{S2})$ be binary valued variables indicating if the stations S1 and S2 are within a 0.5 kilometer radius and within the same cluster, respectively. The relationships are captured with the following rules:

$$w : \text{ADJ}(\texttt{S1}, \texttt{S2}) \wedge \text{DEPART}(\texttt{S1}, \texttt{T}) \quad (21)$$

$$\rightarrow \text{DEPART}(\texttt{S2}, \texttt{T}) \quad (22)$$

$$w : \text{CLUSTER}(\texttt{S1}, \texttt{S2}) \wedge \text{DEPART}(\texttt{S1}, \texttt{T}) \quad (23)$$

$$\rightarrow \text{DEPART}(\texttt{S2}, \texttt{T}) \quad (24)$$

If the $\text{ADJ}(\texttt{S1}, \texttt{S2})$ or $\text{CLUSTER}(\texttt{S1}, \texttt{S2})$ relation is symmetric, then these rules will forecast the demand at the two stations in the ground rules to be nearly equal.

Commuting patterns are also captured via collective modeling. Suppose a station $\texttt{S1}$ is forecasted to have a high arrival rate at a time $\texttt{T}$. If we observe that $\texttt{S2}$ is a common source for $\texttt{S1}$, then we can infer that $\texttt{S2}$ should have a higher departure rate at a time $\texttt{T}$. The inverse dependency is also true. This relation is captured by the rules:

$$w : \text{DEST}(\texttt{S1}, \texttt{S2}) \wedge \text{DEPART}(\texttt{S1}, \texttt{T}) \quad (25)$$

$$\rightarrow \text{ARRIVE}(\texttt{S2}, \texttt{T}) \quad (26)$$

$$w : \text{SOURCE}(\texttt{S1}, \texttt{S2}) \wedge \text{ARRIVE}(\texttt{S1}, \texttt{T}) \quad (27)$$

$$\rightarrow \text{DEPART}(\texttt{S2}, \texttt{T}) \quad (28)$$

The atoms $\text{DEST}(\texttt{S1}, \texttt{S2})$ and $\text{SOURCE}(\texttt{S1}, \texttt{S2})$ represent whether station $\texttt{S2}$ is among the top destinations and sources for the station $\texttt{S1}$, respectively.

5.4. Regularizers

In general, both arrival and departure demand at any given station will tend to be low. We include this notion as a prior by adding the following rules to our PSL model:

$$w : \neg\text{ARRIVE}(\texttt{S}, \texttt{T}) \quad (29)$$

$$w : \neg\text{DEPART}(\texttt{S}, \texttt{T}) \quad (30)$$

Intuitively, negative prior rules can be interpreted as encouraging PSL to make predictions near 0. More formally, the rules result in a regularized MAP inference objective function.

6. Experiments

We evaluate our proposed framework on station level hourly forecasts for both arrival and departure demands. We compare two approaches: autoregressive rules with exogenous information (**AR+X**) and a collective model that combines autoregressive rules, collective rules, exogenous information, and regularizers (**Collective**). Furthermore, both incremental (**INC**) and batch (**BATCH**) multi-step forecasting methods, discussed in Section 5.1, are examined for predicting the next 24 hours of demand.

The **AR+X** model captures dependencies on 1, 2, 12, 24, and 168 hour lags, motivated by Figure 2. The weights for the autoregressive rules were determined via a minimum square error fit on an observed set of the demand time series. Moreover, a restricted search over the weight values of 0.01 and 0.001 was performed on the exogenous rules and we report the best performing configuration. The **INC-AR+X** model reproduces the forecast of a traditional seasonal autoregressive model.

The **Collective** model extends **AR+X** with rules to capture dependencies identified via clustering, spatial distance, and commuting relations. Clustering is performed using Scikit-learn's hierarchical agglomerative clustering algorithm over every stations' demand series. For the spatial rule, two stations are adjacent if they are within a 0.5 kilometer radius and for the commuting patterns, the top three sources and destinations for each station are considered. Again, a restricted search over configurations where rule clusters were assigned weights of either 0.01 or 0.001 was performed for the rule weights. We also analyze the performance of collective models created by adding one relational rule at a time to the **AR+X** model, these models are labeled *Cluster*, *Spatial*, and *Source-Dest*.

All models are evaluated on a consecutive 30 days of 24 hour multi-step forecasting. Demand time series are scaled to the range $[0, 1]$ by dividing 6 times the station's standard deviation and re-scaled for evaluation. Table 3 shows the mean and median daily RMSE and R^2 of all the methods. The metric mean and medians are computed over both arrival and departure demands for most stations in the network. We found some stations to be extreme outliers (perhaps due to outages) skewing both metrics. Thus, if no trips departed from or arrived at a station on a particular day, evaluations of its forecast on that day are omitted.

The **collective** model improves on AR+X in mean and median R^2 and RMSE in both **INC** and **BATCH** forecasting on all three datasets. Though some differences are not visible in the table due to rounding, a paired t-test of the per-day means/medians of these metrics indicate that the improvement is statistically significant. Collective reasoning is most effective on the Boston network where the correlation signal from the collective relationships is strongest (see Table 2). Moreover, sources of relational information in isolation appear to have little effect on performance. In fact, in some cases the models do worse than pure **AR+X**. However, in conjunction, collective information consistently provides performance gains.

Lastly, we see that **Batch** forecasting provides a substantial improvement in forecast quality over **Inc** forecasting on all datasets and models. The performance

		Boston				Bay Area				Los Angeles			
		R2 Mean	R2 Med.	RMSE Mean	RMSE Med.	R2 Mean	R2 Med.	RMSE Mean	RMSE Med.	R2 Mean	R2 Med.	RMSE Mean	RMSE Med.
INC	AR+X	-0.17	0.11	1.33	1.18	-0.34	0.10	0.97	0.85	-0.34	-0.06	0.64	0.57
	Cluster	-0.19	0.10	1.34	1.19	-0.34	0.10	0.98	0.85	-0.33	-0.06	0.64	0.57
	Spatial	-0.20	0.10	1.34	1.19	-0.34	0.10	0.98	0.85	-0.35	-0.07	0.65	0.58
	Source-Dest.	-0.20	0.10	1.34	1.19	-0.34	0.10	0.98	0.85	-0.35	-0.07	0.65	0.58
	Collective	-0.16	0.12	1.32	1.18	-0.32	0.11	0.97	0.85	-0.30	-0.05	0.64	0.57
BATCH	Cluster	**-0.10**	0.15	1.31	1.17	**-0.20**	**0.15**	0.96	0.82	**-0.22**	**-0.02**	0.63	0.56
	Spatial	**-0.11**	0.15	1.31	1.17	**-0.21**	**0.15**	0.96	0.82	**-0.23**	-0.02	0.63	0.56
	Source-Dest.	**-0.11**	0.15	1.31	1.17	**-0.21**	**0.15**	0.96	0.82	**-0.23**	**-0.02**	0.63	0.56
	Collective	**-0.08**	**0.16**	**1.30**	**1.16**	**-0.19**	**0.16**	**0.96**	**0.82**	**-0.21**	**-0.01**	**0.62**	**0.55**

Figure 3: Mean and median daily R^2 and RMSE for 30 consecutive days of hourly station level demand forecasts.

gain of **Batch** over **Inc** forecasting can be explained by the power of collective prediction. **Batch** forecasts are made for a full 24 hour period for all of the stations, allowing the model to make a prediction that most satisfies the rules presented in Section 5. On the other hand, **Inc** forecasts are made incrementally, 1 hour at a time for all the stations. The **Inc** forecasts are also maximizing the satisfaction of the rules in the model, however the incremental forecasting process is akin to making a greedy approximation. This surprising outcome is made possible by scalability of MAP inference in PSL.

7. Conclusion

Shared mobility services, such as bicycle sharing services, are likely to continue to grow in significance, necessitating research that addresses the significant and novel challenges associated with their operation. Service operators need high-quality demand forecasts to support a variety of important decisions, including when to rebalance bicycles to avoid service failures. In this work, we leverage the relational structure in bicycle sharing networks. In doing so, we introduce a highly scalable, efficient framework for effectively unifying a heterogeneous array of signals to extend classical autoregressive time-series modelling. Our framework is able to effectively reason with historical observations, relational information, and exogenous information. In an experimental evaluation, our models achieve more accurate forecasts than standard autoregressive models. Future directions for this research include exploring better weight learning techniques, experimenting with other sources of relational and exogenous information, and extending our models to additional demand forecasting domains.

References

[1] Vitoria Albuquerque, Miguel Sales Dias, and Fernando Bacao. Machine learning approaches to bike-sharing systems: A systematic literature review. *EAI Endorsed Transaction on Smart Cities*, 10, 2021.

[2] Stephen H. Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. Hinge-loss markov random fields and probabilistic soft logic. *Journal of Machine Learning Research*, pages 1–67, 2017.

[3] Breno Beirigo, Frederik Schulte, and Rudy Negenborn. A learning-based optimization approach for autonomous ridesharing platfoorms with service-level contract and on-demand hiring of idle vehicles. *Transportation Science*, 2022.

[4] Mike Benchimol, Pascal Benchimol, Benoit Chappert, Arnaud De La Taille, Fabien Laroche, Frederic Meunier, and Ludovic Robinet. Balancing the stations of a self service ”bike hire” system. *RAIRO-Operations Research*, 45:37–61, 2011.

[5] Bluebikes. Bluebikes bike share. `https://www.bluebikes.com/system-data`, 2022. Accessed: 2022-08-28.

[6] George Box, Gwilym Jenkins, Gregory Reinsel, and Greta Ljung. *Time Series Analysis: Forecasting and Control*. Wiley, 2015.

[7] Luca Cagliero, Tania Cerquitelli, Silvia Chiusano, Paolo Garza, and Xin Xiao. Predicting critical conditions in bicycle sharing systems. *Computing*, 99(1):39–57, 2017.

[8] Boyd Cohen and Jan Kietzmann. Ride on! mobility business models for the sharing economy. *Organization and Environment*, 27(3):2779–296, 2014.

[9] Charles Dickens, Connor Pryor, Eriq Augustine, Alex Miller, and Lise Getoor. Context-aware online collective inference for templated graphical models. In *International Conference on Machine Learning (ICML)*, 2021.

[10] Ezgi Eren and Volkan Emre UZ. A review on bike-sharing: The factors affecting bike-sharing demand. *Sustainable Cities and Society*, 54, 2020.

[11] Ahmadreza Faghih-Imani and Naveen Eluru. Incorporating the impact of spatio-temporal interactions on bicycle sharing system demand: A case study of New York CitiBike system. *Journal of Transport Geography*, 54:218–227, 2016.

[12] Daniel Freund, Ashkan Norouzi-Fard, Alice Paul, Carter Wang, Shane Henderson, and David Shmoys. *Data-Driven Rebalancing Methods for Bike-Share Systems*, pages 255–278. 03 2020.

[13] Pierre Hulot, Daniel Aloise, and Sanjayy Dominik Jena. Toward station-level demand prediction for effective rebalancing in bike-sharing systems. In *International Conference on Knowledge Discovery & Data Mining (SIGKDD)*, 2018.

[14] Rob Hyndman and George Athanasopoulos. *Forecasting Principles and Practice*. OTexts, 2021.

[15] Ashish Kabra, Elena Belavina, and Karan Girotra. Bike-sharing systems: Accessibility and availability. *Chicago Booth Research*, 4(15), 2018.

[16] Fabio Kon, Ederson Cassio Ferreira, Higor Amario de Souza, Fabio Duarte, Paolo Santi, and Carlo Ratti. Abstracting mobility flows from bike-sharing systems. *Public Transport*, 2021.

[17] Denis Kwiatkowski, Peter C.B. Phillips, Peter Schmidt, and Yongcheol Shin. Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of Econometrics*, 54(1):159–178, 1992.

[18] Gilbert Laporte, Frédéric Meunier, and Roberto Wolfler Calvo. Shared mobility systems. *Annals of Operations Research*, 271(4):105–126, 2018.

[19] Xiaofeng Li, Yao-Jan Wu, and Alireza Khani. Investigating a small-sized bike-sharing system's impact on transit usage: A synthetic control analysis in tucson, arizon. *Public Transport*, 2022.

[20] Lei Lin, Zhengbing He, and Srinivas Peeta. Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach. *Transportation Research Part C: Emerging Technologies*, 97:258–276, 2018.

[21] Metro Bikes. Metro bike share. `https://bikeshare.metro.net/about/data`, 2022. Accessed: 2022-08-28.

[22] Metropolitan Transportation Commission. Bay area bike share. `https://mtc.ca.gov/our-work/operate-coordinate/traveler-services/bay-area-bike-share`, 2022. Accessed: 2022-08-28.

[23] Eoin O'Mahony and David Smhmoys. Data analysis and optimization for (citi)bike sharing. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2015.

[24] Jay Pujara, Ben London, and Lise Getoor. Budgeted online collective inference. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2015.

[25] Christian Rudloff and Bettina Lackner. Modeling demand for bikesharing systems: Neighboring stations as source for demand and reasoning for structural breaks. *Transportation Research Record*, 2430(1):1–11, 2014.

[26] Simon Ruffieux, Elena Mugellini, and Omar Abou Khaled. Bike usage forecasting for optimal rebalancing operations in bike-sharing systems. In *International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 854–858, 2018.

[27] J Schuijbroek, R.C Hampshire, and W.-J van Hoeve. Inventory rebalancing and vehicle routing in bike sharing systems. *European Journal of Operational Research*, 257(3):992–1004, 2017.

[28] Chenchen Xu, Junyi Ji, and Pan Liu. The station-free sharing bike demand forecasting with a deep learning approach and large-scale datasets. *Transportation Research Part C: Emerging Technologies*, 2018.

[29] Ying Yu, Wenjie Yi, Yuanyue Feng, and Jia Liu. Understanding the intention to use commerical bike-sharing systems: An integration of tam and tpb. In *Hawaii International Conference on Systems Sciences (HICSS)*, 2018.