Spatio-Temporal Random Fields: Compressible Representation and Distributed Estimation

Nico Piatkowski, Sangkyun Lee and Katharina Morik
TU Dortmund University, Dortmund 44227, Germany
{nico.piatkowski,sangkyun.lee,katharina.morik}@tu-dortmund.de

Abstract

Modern sensing technology allows us enhanced monitoring of dynamic activities in business, traffic, and home, just to name a few. The increasing amount of sensor measurements, however, brings us the challenge for efficient data analysis. This is especially true when sensing targets can interoperate – in such cases we need learning models that can capture the relations of sensors, possibly without collecting or exchanging all data. Generative graphical models namely the Markov random fields (MRFs) fit this purpose, which can represent complex spatial and temporal relations among sensors, producing interpretable answers in terms of probability. The only drawback will be the cost for inference, storing and optimizing a very large number of parameters – not uncommon when we apply them for real-world applications.

In this paper, we investigate how we can make discrete probabilistic graphical models practical for predicting sensor states in a spatio-temporal setting. A set of new ideas allows keeping the advantages of such models while achieving scalability. We first introduce a novel alternative to represent model parameters, which enables us to compress the parameter storage by removing uninformative parameters in a systematic way. For finding the best parameters via maximal likelihood estimation, we provide a separable optimization algorithm that can be performed independently in parallel in each graph node. We illustrate that the prediction quality of our suggested methods is comparable to those of the standard MRFs and a spatio-temporal k-nearest neighbor method, while using much less computational resources.

1 Introduction

Sensor-based monitoring and prediction has become a hot topic in a large variety of applications. According to the slogan *Monitor, Mine, Manage* [1], series of data from heterogeneous sources are to be put to good use in diverse applications. A view of data mining towards *distributed sensor measurements* is presented in the book on ubiquitous knowledge discovery [11]. There are several approaches to distributed stream mining based on work like, e.g., Wolff *et al.* [21] or Sagy *et al.* [15]. The goal in these approaches is a general model (or function) which is built on the basis of local models while restricting communication costs. Most often, the global model allows to answer threshold queries, but also clustering of nodes is sometimes handled. Although the function is more complex, the model is global and not tailored for the prediction of measurements at a particular location. In contrast, we want to predict some sensor's state at some point in time given relevant previous and current measurements of itself and other sensors.

Since his influential book, David Luckham has promoted *complex event processing* successfully [9]. Detecting events in streams of data has accordingly been modeled, e.g. in the context of monitoring hygiene in a hospital [18]. However, in our case, the monitoring does not imply certain events. We do not aim at finding patterns that define an event, although they may show up as a side effect. We rather want to predict a certain state at a particular sensor or set of sensors taking into account the context of other locations and points in time. Although related, the tasks differ.

Let us illustrate the task of *spatio-temporal state prediction* by an example from traffic modeling. The structure of the model is given by a street network, which represents spatial relationships. Nodes within the network represent places, where the traffic is measured over time. The state of a node is the congestion at this street segment. At training time, we do not know which place at which time needs to be predicted as "jam". Given observations of the state variables at the nodes, a model is trained. The model must answer queries for all parts of the network and all points in time. For example:

- Given the traffic densities of all roads in a street network at discrete time points t_1, t_2, t_3 (e.g., Monday, Tuesday, Wednesday 8 o'clock): indicate the probabilities of traffic levels on a particular road A at another time point, not necessarily following the given ones (e.g., Thursday 7 o'clock).

One particular interest lies in learning probabilistic models for answering such queries in resource constrained environments. This addresses huge amounts of data on quite fast compute facilities as well as a rather moderate data volume on embedded or ubiquitous devices.

1.1 Previous Work

In this section, an overview of previous contributions to spatio-temporal modeling is given. The task of *traffic forecasting* is often solved by simulations [10]. This presupposes a model instead of learning it. In the course of urban traffic control, events are merely propagated that are already observed, e.g., a jam at a particular highway section results in a jam at another highway section, or the prediction is based on a physical rule that predicts a traffic jam based on a particular congestion pattern [3]. Many approaches apply statistical time series methods like auto-regression and moving average [20]. They do not take into account spatial relations but restrict themselves to the prediction of the state at one location given a series of observations at this particular location. An early approach is presented by Whittaker *et al.* [19], using a street network topology that represents spatial relations. The training is done using simply Kalman filters, which is not as expressive as is necessary for queries like the ones above. A statistical relational learning approach to traffic forecasting uses explicit rules for modeling spatio-temporal dependencies like $congestion(+s_1, h) \wedge next(s_1, s_2) \Rightarrow congestion(+s_2, h+1)$ [8]. Training is done by a Markov Logic Network delivering conditional probabilities of congestion classes. The discriminative model is restricted to binary classification tasks and the spatial dependencies need to be given by hand-tailored rules. Moreover, the model is not sparse and training is not scaleable. Even for a small number of sensors, training takes hours of computation. When the estimation of models for spatio-temporal data on ubiquitous devices is considered, e.g. learning to predict smartphone usage patterns based on time and visited places, minutes are the order of magnitude in demand. Hence, also this advanced approach does not yet meet the demands of the spatio-temporal prediction task in resource constrained environments.

Some geographically weighted regression or non-parametric k-Nearest Neighbour (kNN) methods model a task similar to spatio-temporal state prediction [23, 12]. The regression expresses the temporal dynamics and the weights express spatial distances. Another way to introduce the spatial relations into the regression is to encode the spatial network into a kernel function [7]. The kNN method by Lam *et al.* [6] models correlations in spatio-temporal data not only by their spatial but also by their temporal distance. As stated for spatio-temporal state prediction task, the particular place and time in question need not be known in advance, because the lazy learner kNN determines the prediction at question-time. However, also this approach does not deliver probabilities along with the predictions. For some applications, for instance, traffic prognoses for car drivers, a probabilistic assertion is not necessary. However, in applications of disaster management, the additional information of likelihood is wanted.

As is easily seen, generative models fit the task of spatio-temporal state prediction. For notational convenience, let us assume just one variable x. The *generative model* $p(x, y)$ allows to derive $p(y|x) = p(x, y)/p(x)$ as well as $p(x|y) = p(x, y)/p(y)$. In contrast, the *discriminative model* $p(y|x)$ must be trained specifically for each y. In our example, for each place, a distinct model would need to be trained. Hence, a huge set of discriminative models would be necessary to express one generative model. A discussion of discriminative versus generative models can be found in a study by Ng and Jordan [13]. Here, we refer to the capability of interpolation (e.g., between points in time) of generative models and their informativeness in delivering probability estimates instead of mere binary decisions.

Spatial relations are naturally expressed by *graphical models*. For instance, discriminative graphical models – as are Conditional Random Fields (CRFs) – have been used for object recognition over time [2], but also generative graphical models such as Markov Random Fields (MRFs) have been applied to video or image data [22, 4]. The number of training instances does not influence the model complexity of MRFs. However, the number of parameters can exceed millions easily. In particular when using MRFs for spatio-temporal state prediction, the many spatial and temporal relations soon lead to inefficiency.

1.2 Graphical Models

The formalism of probabilistic graphical models provides a unifying framework for capturing complex dependencies among random variables, and building large-scale multivariate statistical models [17]. Let $G = (V, E)$ be an undirected graph with the set of vertices V and the set of edges $E \subset V \times V$. For each node (or vertex) $v \in V$, let X_v be a random variable, taking values x_v in some space $\mathcal{X}_v$. The concatenation of all $n = |V|$ variables yields a multivariate random variable $\boldsymbol{X}$ with state space $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n$. Training delivers a full probability distribution over the random variable $\boldsymbol{X}$. Let $\boldsymbol{\phi}$ be an indicator function or *sufficient statistic* that indicates if a configuration $\boldsymbol{x}$ obeys a certain event $\{\boldsymbol{X}_\alpha = \boldsymbol{x}_\alpha\}$ with $\alpha \subseteq V$. We use the short hand notation $\{\boldsymbol{x}_\alpha\}$ to denote the event $\{\boldsymbol{X}_\alpha = \boldsymbol{x}_\alpha\}$. The functions of $\boldsymbol{x}$ defined in the following can be also considered as functions of $\boldsymbol{X}$ – we replace $\boldsymbol{x}$ by $\boldsymbol{X}$ when it makes their meaning clearer. Restricting α to vertices and edges, one gets

$$\boldsymbol{\phi}_{\{v=x\}}(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x}_v = x \\ 0 & \text{otherwise,} \end{cases}$$

$$\boldsymbol{\phi}_{\{(v,w)=(x,y)\}}(\boldsymbol{x}) = \begin{cases} 1 & \text{if } (\boldsymbol{x}_v, \boldsymbol{x}_w) = (x, y) \\ 0 & \text{otherwise} \end{cases}$$

with $\boldsymbol{x} \in \mathcal{X}$, $\boldsymbol{x}_v \in \mathcal{X}_v$ and $y \in \mathcal{X}_w$. Let us now define vectors for collections of those indicator functions:

$$\begin{aligned} \boldsymbol{\phi}_v(\boldsymbol{x}) &:= \left[\boldsymbol{\phi}_{\{v=x\}}(\boldsymbol{x})\right]_{x \in \mathcal{X}_v}, \\ \boldsymbol{\phi}_{(v,w)}(\boldsymbol{x}) &:= \left[\boldsymbol{\phi}_{\{(v,w)=(x,y)\}}(\boldsymbol{x})\right]_{(x,y) \in \mathcal{X}_v \times \mathcal{X}_w}, \\ \boldsymbol{\phi}(\boldsymbol{x}) &:= \left[\boldsymbol{\phi}_v(\boldsymbol{x}), \boldsymbol{\phi}_e(\boldsymbol{x}) : \forall v \in V, \forall e \in E\right]. \end{aligned} \tag{1}$$

The vectors are constructed for fixed but arbitrary orderings of V, E and $\mathcal{X}$. The dimension of $\boldsymbol{\phi}(\boldsymbol{x})$ is thus $d = \sum_{v \in V} |\mathcal{X}_v| + \sum_{(v,u) \in E} |\mathcal{X}_v| \times |\mathcal{X}_u|$. Now, consider a data set $\mathcal{D} = \left\{\boldsymbol{x}^1, \boldsymbol{x}^2, \ldots, \boldsymbol{x}^N\right\}$ with instances $\boldsymbol{x}^i$. Each $\boldsymbol{x}^i$ consists of an assignment to every node in the graph. It defines a full joint state of the random

variable $\boldsymbol{X}$. The quantities

$$\hat{\boldsymbol{\mu}}_{\{v=x\}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\phi}_{\{v=x\}}(\boldsymbol{x}^i),$$

$$\hat{\boldsymbol{\mu}}_{\{(v,w)=(x,y)\}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\phi}_{\{(v,w)=(x,y)\}}(\boldsymbol{x}^i)$$

are known as *empirical moments* and they reflect the empirical frequency estimates of the corresponding events. We say that a given probability mass function p with base measure ν and expectations $\mathbb{E}_p\left[\boldsymbol{\phi}_{\{\boldsymbol{x}_\alpha\}}(\boldsymbol{x})\right]$ is *locally consistent* with data $\mathcal{D}$ if and only if p satisfies the *moment matching condition*

$$\mathbb{E}_p\left[\boldsymbol{\phi}_{\{\boldsymbol{x}_\alpha\}}(\boldsymbol{x})\right] = \hat{\boldsymbol{\mu}}_{\{\boldsymbol{x}_\alpha\}}, \forall \alpha \in V \cup E,$$

i.e. the density p is consistent with the data w.r.t. the empirical moments. This problem is underdetermined, in that there are many densities p that are consistent with the data, so that we need a principle for choosing among them. The principle of maximum entropy is to choose, from among the densities consistent with the data, the densities p^* whose *Shannon entropy* $\mathcal{H}(p)$ is maximal. It can be shown that the optimal solution p^* takes the form of an exponential family of densities

$$p_{\boldsymbol{\theta}}(\boldsymbol{X} = \boldsymbol{x}) = \exp[\langle \boldsymbol{\theta}, \boldsymbol{\phi}(\boldsymbol{x})\rangle - A(\boldsymbol{\theta})],$$

parametrized by a vector $\boldsymbol{\theta} \in \mathbb{R}^d$. Note that the parameter vector $\boldsymbol{\theta}$ and the sufficient statistics vector $\boldsymbol{\phi}(\boldsymbol{x})$ have the same length d. The term $A(\boldsymbol{\theta})$ is called the *log partition function*,

$$A(\boldsymbol{\theta}) := \log \int_{\mathcal{X}} \exp[\langle \boldsymbol{\theta}, \boldsymbol{\phi}(\boldsymbol{x})\rangle]\nu(d\boldsymbol{x}),$$

which is defined with respect to a reference measure $d\nu$ such that $P[X \in S] = \int_S p_{\boldsymbol{\theta}}(x)\nu(dx)$ for any measurable set S. Expanding $\boldsymbol{\phi}(\boldsymbol{x})$ by means of (1) reveals the usual density of pairwise undirected graphical models, also known as *pairwise Markov random field*

$$p_{\boldsymbol{\theta}}(\boldsymbol{X} = \boldsymbol{x}) = \frac{1}{\Psi(\boldsymbol{\theta})}\prod_{v\in V}\psi_v(\boldsymbol{x})\prod_{(v,w)\in E}\psi_{(v,w)}(\boldsymbol{x}).$$

Here, $\Psi = \exp A$ is the cumulant-generating function of $p_{\boldsymbol{\theta}}$, and ψ_α are the so-called *potential functions*.

If the data set contains solely fully observed instances, the parameters may be estimated by the maximum likelihood principle. The estimation of parameters in the case of partially unobserved data is a challenging topic on its own. Here, we assume that the data set $\mathcal{D}$ contains only fully observed instances. The *likelihood* $\mathcal{L}$ and the *average log-likelihood* ℓ of parameters $\boldsymbol{\theta}$ given a set of i.i.d. data $\mathcal{D}$ are defined as

$$\mathcal{L}(\boldsymbol{\theta}; \mathcal{D}) := \prod_{i=1}^{N} p_{\boldsymbol{\theta}}(\boldsymbol{x}^i) \quad \text{and}$$

$$\ell(\boldsymbol{\theta}; \mathcal{D}) := \frac{1}{N}\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^i) = \langle \boldsymbol{\theta}, \hat{\boldsymbol{\mu}}\rangle - A(\boldsymbol{\theta}).$$

The latter is usually maximized due to numerical inconveniences of $\mathcal{L}$.

2 From Linear Chains to Spatio-Temporal Models

Sequential undirected graphical models, also known as linear chains, are a popular method in the natural language processing community [5, 16]. There, consecutive words or corresponding word features are connected to a sequence of labels that reflects an underlying domain of interest like entities or part of speech tags. If we consider a sensor network G that generates measurements over space as a word, then it would be appealing to think of the instances of G at different timepoints, like words in a sentence, to form a temporal chain $G_1 - G_2 - \cdots - G_T$. We will now present a formalization of this idea followed by some obvious drawbacks. Afterwards we will discuss how to tackle those drawbacks and derive a tractable class of generative graphical models for the spatio-temporal state prediction task.

We first define the part of the graph corresponding to the time t as the *snapshot graph* $G_t = (V_t, E_t)$, for $t = 1, 2, \ldots, T$. Each snapshot graph G_t replicates a given *spatial graph* $G_0 = (V_0, E_0)$, which represents the underlying physical placement of sensors, i.e., the spatial structure of random variables that does not change over time. We also define the set of spatio-temporal edges $E_{t-1;t} \subset V_{t-1} \times V_t$ for $t = 2, \ldots, T$ and $E_{0;1} = \emptyset$, that represent dependencies between adjacent snapshot graphs G_{t-1} and G_t, assuming a Markov property among snapshots, so that $E_{t;t+h} = \emptyset$ whenever $h > 1$ for any t. Note that the actual time gap between any two time frames t and $t+1$ can be chosen arbitrarily.

The entire graph, denoted by G, consists of the snapshot graphs G_t stacked in order for time frames $t = 1, 2, \ldots, T$ and the temporal edges connecting them: $G := (V, E)$ for $V := \cup_{t=1}^{T} V_t$ and $E := \cup_{t=1}^{T}\{E_t \cup E_{t-1;t}\}$.

The spatial graph G_0 and the sizes of the vertex state spaces $\mathcal{X}_v$ determine the number of model parameters d. In order to compute this quantity, we consider the exemplary construction of G. from G_0. First, all vertices v and all edges (u, v) from G_0 are copied exactly T times and added to $G = (V, E)$, whereas each copy is indexed by time t, i.e. $v \in V_0 \Rightarrow v_t \in V, 1 \le t \le T$ and likewise for the edges. Then, for each vertex $v_t \in V$ with $t \le T-1$, a temporal edge (v_t, v_{t+1}) is added to G. Finally, for each edge $(v_t, u_t) \in E$ with $t \le T-1$, the two spatio-temporal edges (v_t, u_{t+1}) and (v_{t+1}, u_t) are also added to G. The number of parameters per vertex v is $|\mathcal{X}_v|$ and accordingly $|\mathcal{X}_v||\mathcal{X}_u|$ per edge (v, u). If we assume that all vertices $v, u \in V$ share a common state space and that state spaces do not change over time, i.e. $\mathcal{X}_{v_t} = \mathcal{X}_{u_{t'}}, \forall v, u \in V, 1 \le t, t' \le T$, the total number of parameters is

$$d = T|V_0|\,|\mathcal{X}_{v_t}| + \left[(T-1)(|V_0| + 3|E_0|) + |E_0|\right]|\mathcal{X}_{v_t}|^2$$

with some arbitrary but fixed vertex v_t. Note that the last two assumptions are only needed to simplify the computation of d, the spatio temporal random field that is described in the following section is not restricted by any of these assumptions.

This model now truly expresses temporal and spatial relations between all locations and points in time for all features. However, the memory requirements of

such models are quite high due to the large problem dimension. Even loading or sending models may cause issues when mobile devices are considered as a platform. Furthermore, the training does not scale well because of stepsize adaption techniques that are based on sequential (i.e., non-parallel) algorithms.

The derivation and empirical evaluation of the compressible representation and distributed estimation can be found in [14].

Acknowledgments

Work on this paper has been supported by Deutsche Forschungsgemeinschaft (DFG) within the Collaborative Research Center SFB 876 "Providing Information by Resource-Constrained Analysis", projects A1 and C1.

References

[1] David Campbell. Is it still Big Data if it fits in my pocket? In *Proceedings of the VLDB Endowment*, volume 4, page 694, 2011.

[2] Bertrand Douillard, Dieter Fox, and Fabio T. Ramos. A spatio-temporal probabilistic model for multi-sensor object recognition. In *IEEE/RSJ International Conference on IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2402–2408, 2007.

[3] S. F. Hafstein, R. Chrobok, A. Pottmeier, and M. Schreckenbergand F. Mazur. A high-resolution cellular automata traffic simulation model withapplication in a freeway traffic information system. *Computer-Aided Civil and Infrastructure Engineering*, 19(5):338–350, 2004.

[4] Rui Huang, Vladimir Pavlovic, and Dimitris Metaxas. A new spatio-temporal mrf framework for video-based object segmentation. In *The 1st International Workshop on Machine Learning for Vision-based Motion Analysis*, 2008.

[5] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning*, pages 282–289, 2001.

[6] W. H. K. Lam, Y. F. Tang, and M. Tam. Comparison of two non-parametric models for daily traffic forecasting in hong kong. *Journal of Forecasting*, 25(3):173–192, 2006.

[7] Thomas Liebig, Zhao Xu, Michael May, and Stefan Wrobel. Pedestrian quantity estimation with trajectory patterns. In *Machine Learning and Knowledge Discovery in Databases*, volume 7524 of *Lecture Notes in Computer Science*, pages 629–643. Springer, 2012.

[8] Marco Lippi, Matteo Bertini, and Paolo Frasconi. Collective traffic forecasting. In *Machine Learning and Knowledge Discovery in Databases*, volume 6322 of *Lecture Notes in Computer Science*, pages 259–273. Springer, 2010.

[9] David Luckham. *The Power of Events - An Introduction to Complex Event Processing in Distributed Enterprise Systems*. Addison Wesley, 2002.

[10] Sigurdur F. Marinosson, Roland Chrobok, Andreas Pottmeier, Joachim Wahle, and Michael Schreckenberg. Simulation framework for the autobahn traffic in North Rhine-Westphalia. In *Cellular Automata – 5th Int. Conf. on Cellular Automata for Research and Industry*, pages 2977–2980. Springer, 2002.

[11] Michael May and Lorenza Saitta, editors. *Ubiquitous Knowledge Discovery*, volume 6202 of *Lecture Notes in Artificial Intelligence*. Springer, 2010.

[12] M. May, D. Hecker, C. Körner, S. Scheider, and D. Schulz. A vector-geometry based spatial knn-algorithm for traffic frequency predictions. *Data Mining Workshops, International Conference on Data Mining*, 0:442–447, 2008.

[13] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in Neural Information Processing Systems*, 14:841–848, 2002.

[14] Nico Piatkowski, Sangkyun Lee, and Katharina Morik. Spatio-temporal random fields: Compressible representation and distributed estimation. *Machine Learning Journal*, 93(1):115–139, 2013 2013.

[15] Guy Sagy, Daniel Keren, Izchak Sharfman, and Assaf Schuster. Distributed threshold querying of general functions by a difference of monotonic representation. In *Proceedings of the VLDB Endowment*, volume 4, 2011.

[16] Charles Sutton and Andrew McCallum. An introduction to conditional random fields for relational learning. In *Introduction to Statistical Relational Learning*. MIT Press, 2007.

[17] Martin J. Wainwright and Michael I. Jordan. Graphical Models, Exponential Families, and Variational Inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2007.

[18] Di Wang, Elke A. Rundensteiner, and Richard T. Ellison. Active complex event processing of event streams. In *Procs. of the VLDB Endowment*, volume 4, 2011.

[19] Joe Whittaker, Simon Garside, and Karel Lindveld. Tracking and predicting a network traffic process. *International Journal of Forecasting*, 13(1):51–61, March 1997.

[20] B.M. Williams and L.A. Hoel. Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results. *Journal of Transportation Engineering*, 129(6):664–672, 2003.

[21] Ran Wolff, Kanishka Badhuri, and Hillol Kargupta. A generic local algorithm for mining data streams in large distributed systems. *IEEE Transactions on Knowledge and Data Engineering*, 21(4):465–478, 2009.

[22] Zhaozheng Yin and Robert Collins. Belief propagation in a 3D spatio-temporal MRF for moving object detection. *IEEE Computer Vision and Pattern Recognition*, 2007.

[23] F. Zhao and N. Park. Using geographically weighted regression models to estimate annual average daily traffic. *Journal of the Transportation Research Board*, 1879(12):99–107, 2004.