

Spatio-Temporal Models For Sustainability

Nico Piatkowski
Department of Computer
Science, LS 8
TU Dortmund University
Dortmund, Germany
nico.piatkowski@tu-
dortmund.de

Sangkyun Lee
Department of Computer
Science, LS 8
TU Dortmund University
Dortmund, Germany
sangkyun.lee@tu-
dortmund.de

Katharina Morik
Department of Computer
Science, LS 8
TU Dortmund University
Dortmund, Germany
katharina.morik@tu-
dortmund.de

ABSTRACT

Many applications that aim at enhancing sustainability rely on some sort of spatio-temporal model. The task can be monitoring or prediction in traffic networks, power grids, building energy management, river flow volume, and sea level – to mention just a few. The positive effect on the environment is achieved by a better control, better planning of processes or better disaster management. Spatio-temporal models predict some states of variables over time that are spatially ordered in some topology. They can be used in a variety of applications, ranging from energy-saving technological administration to monitoring for early alarms or to supporting better emergency plans. Graphical Models have been successfully used for predictions based on structured data. Here, we introduce a functional form for the model parameters that allows a spatio-temporal predictive analysis with continuous time. Though intended also to be used in engineering approaches, there, we are missing public data sets. Hence, the use of our model is demonstrated on two exemplary applications, namely the prediction of sea-levels for tsunami prediction and the prediction of traffic jam as a consequence of a flow of refugees in case of a nuclear accident.

Categories and Subject Descriptors

V.2 [Transportation]; H.3 [Mining massive throughput sensor streams]

General Terms

Spatio-Temporal Models, Graphical Models

1. INTRODUCTION

Sustainability applications [15] and data mining challenges concerning climate change [4] often cope with spatio-temporal data. For instance, the sea level data (<http://ngdc.noaa.gov/hazard/recenttsunamis.shtml>) can be used in order to answer queries like:

This is a draft. Permission to make digital or hard copies of portions of this work for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

To appear in ACM SustKDD'12, August 12, 2012, Beijing, China

- Given a high sea level near Shizuoka and near Tokyo at time 1, what will be the level near San Francisco at time t ?

Answers to these questions can be used, e.g., for the tsunami prediction.

Traffic data is important for logistics applications as well as in scenarios of disaster management. Imagine, for instance, that a hazard happens at a large industrial complex. People will want to move away very fast and transporting their minimal goods. The resulting traffic volume demands the street network. Disaster management will act on that behalf and this requires some facts. Here, we do not investigate the disaster management but look at questions whose answers might be useful input to the traffic administration. Traffic data can be used to answer queries like:

- Given the traffic of all roads in a street network at time points $1, 2, \dots, t - 1$, what will be the traffic level on road A at time t (or $t + h$)?
- If there is a traffic jam at crossing (A, B) at time t what will be the state of road C at time $t + h$?

Queries need not be restricted to predict the next point in time, but may ask for some point in the future within a certain horizon, h . Our goal is to mine probabilistic answers to those questions from spatio-temporal data.

Related Work. A view of data mining towards *distributed sensor measurements* is presented in the book on ubiquitous knowledge discovery [14]. There are several approaches to distributed stream mining based on work like, e.g., [22] or [18]. 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, our model can predict some sensor's state at some point in time given relevant previous and current measurements of itself and other sensors.

Graphical models based on *frequent set mining* have been used to model spatial movements over time, e.g., [9]. Such

models deliver all subgraphs that are more frequent than a user-given threshold. They do not deliver any probabilistic model, though. Where probabilistic modeling also starts with counting, it does not ignore seldom events. If we consider the minimal support of a subgraph a probability, frequent set mining deletes events with a probability lower than a threshold. It is hard to justify such a decision in cases where the influence of low frequency events cannot be excluded. Moreover, probabilities can be interpolated. A probabilistic model covers the overall space of all possible relations in a continuous way. If we know, for example, the probability of a certain state at 3 o'clock and its probability at 4 o'clock and miss any measurements in between, we can estimate the probability at, say, 3:30 o'clock. In contrast, frequent subgraph mining delivers patterns that do not allow any interpolation.

Since his influential book, David Luckham has promoted *complex event processing* successfully [12]. According to the slogan *Monitor, Mine, Manage* [2], series of data from heterogeneous sources are to be put to good use in diverse applications. Detecting events in streams of data has been modeled, e.g. in the context of monitoring hygiene in a hospital [20]. 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.

One of the tasks by which we illustrate our approach, namely, *predicting motorway traffic*, has been performed using a street network topology, that represents spatial relations [21]. The training, however, was done using simply Kalman filters, which do not allow to answer questions as those listed above. The prediction task of traffic forecasting is often solved using simulations [13]. However, more recent articles report successful learning approaches, e.g. using Markov Logic Networks [10]. We propose a spatio-temporal model which, in contrast to standard MRF, is capable of expressing periodic behavior by assuming temporal identity between future and past events.

Time series mining and summarizing is undoubtedly a hot research topic [1]. Dealing with sea level data, a clustering method has been proposed that forms effective indices for time series of sea surface temperature or sea level pressure [19]. Tracing evolving subspace clusters in temporal climate data reduces the big data by selecting relevant attributes [5]. Another data reduction approach points out, that clustering streaming data requires to ignore some measurements [17]. Again, these approaches do not solve the problem of predicting future measurements at particular locations. Moreover, they don't take into account the spatial relations in the learned model.

Spatial relations are naturally expressed by *graphical models*. For instance, in the course of analyzing video or image data, graphical models such as Markov Random Fields (MRF) have been used [23], [7]. The standard algorithm for training graphical models is Belief Propagation (BP). It converts local parameters into local probabilities which fit nicely to-

gether, that is, making them globally consistent with respect to the topology. BP is used for the distributed computation of the gradient when minimizing the error. Also standard, we use the Graphlab framework [11] for implementing the new spatio-temporal model. This model itself, however, moves beyond standard approaches. Where regular MRF use just one weight per node and point in time (i.e. a linear, pointwise parametrization), the new model, in contrast, fits a function to the points over time (i.e. a compact piecewise parametrization). This enhancement of MRF is one of the contributions of this paper.

Although the here proposed spatio-temporal MRF is highly related to *Dynamic Bayesian Networks* (DBN), DBNs use directed acyclic graphs (DAG) to represent conditional dependencies. In contrast, MRFs are undirected models which allow cyclic probabilistic dependencies among sensor data. In some cases, it might be hard to impose an ordering of the sensors, which is needed to build the conditional dependency structure of DBNs, e.g. in wireless sensor networks. Clearly, if a set of temperature sensors X is located in the same room, their values are likely to depend on each other. This fact can be intuitively encoded into an undirected model by placing edges between nearby sensors. In case of DBNs, a similar graphical model may be built, but it has to contain additional edges in order to encode all possible links of the undirected MRFs dependency structure. Although both models do result in the same joint density $p(X)$, conditional dependencies between sensors that are not neighbors w.r.t. their physical deployment increase the communication overhead if the model is implemented directly into a sensor network. Furthermore, the additional edges induce a higher memory complexity per node. Please note that the same holds if a DAG structure has to be converted to an undirected model. Nevertheless, an undirected model fits naturally to the physical deployment of sensors and no explicit ordering of the sensors has to be chosen.

In this paper, we present a new approach to spatio-temporal graphical models with applications for sustainability. Its base model is described in Section 2. Applications illustrate such models (Section 3). In Section 4, we indicate opportunities for further work.

2. SPATIO-TEMPORAL MODELS

Spatio-temporal models enhance graphical models of discrete random variables whose probability distributions are represented by exponential families. The enhancement can be combined with any kind of a graphical model. Without loss of generality, we use *Markov random fields* (MRF). We use $|S|$ to denote the cardinality of a finite set S . Symbols describing a vector entity are set in bold.

2.1 Graphical Models

MRF are based on a graph $G = (V, E)$ representing the conditional independency structure among nodes (sensors) $v \in V$ by edges (connections) $(v, w) \in E \subset V \times V$. Each node is associated with a discrete random variable $X_v : \Omega_v \rightarrow \mathcal{X}_v$ for an event set Ω_v and a set of states \mathcal{X}_v . All variables are stacked together to form one multivariate random variable \mathbf{X} . A binary vector $\phi(\mathbf{X})$ of length $d > 0$, contains the states of nodes and the states of pairs of connected nodes. The components of this vector for sensors v, w and states

a, b are defined as follows:

$$\begin{aligned}\phi_{v,a}(\mathbf{X}) &:= \begin{cases} 1 & \text{if } X_v = a \\ 0 & \text{otherwise,} \end{cases} \\ \phi_{vw,ab}(\mathbf{X}) &:= \begin{cases} 1 & \text{if } (X_v, X_w) = (a, b) \\ 0 & \text{otherwise.} \end{cases}\end{aligned}\quad (1)$$

The vector is usually quite sparse, because only a rather small number of states of a variable (values of an attribute) are true, most being zero. This definition represents the state of a entire graph G by the binary vector $\phi(\mathbf{X})$. For the ease of notation, but without loss of generality, we use the same set of states \mathcal{X} for all random variables $X_v, v \in V$. In this case the vector $\phi(\mathbf{X})$ is of length $d = |V| \times |\mathcal{X}| + |E| \times |\mathcal{X}|^2$. As an implication of the binary representation, the expectation of a sufficient statistic is its *marginal probability*, that is,

$$\begin{aligned}\mu_{v,a} &:= \mathbb{E}[\phi_{v,a}(\mathbf{X})] = \mathbb{P}[X_v = a], \\ \mu_{vw,ab} &:= \mathbb{E}[\phi_{vw,ab}(\mathbf{X})] = \mathbb{P}[X_v = a, X_w = b].\end{aligned}$$

We can estimate these quantities from historical sensor readings by counting occurrences, e.g. for N historical measurements per sensor, we have

$$\begin{aligned}\tilde{\mu}_{v,a} &:= \tilde{\mathbb{E}}[\phi_{v,a}(\mathbf{X})] = \frac{1}{N} \sum_{i=1}^N \phi_{v,a}(\mathbf{x}^{(i)}), \\ \tilde{\mu}_{vw,ab} &:= \tilde{\mathbb{E}}[\phi_{vw,ab}(\mathbf{X})] = \frac{1}{N} \sum_{i=1}^N \phi_{vw,ab}(\mathbf{x}^{(i)}),\end{aligned}\quad (2)$$

for all sensors v, w and states a, b . As a consequence of the Hammersley-Clifford Theorem [6], each MRF corresponds to an exponential family distribution parametrized by a vector $\theta \in \mathbb{R}^d$, whose density can be written as

$$p_{\theta}(\mathbf{X}) = \exp\{\langle \theta, \phi(\mathbf{X}) \rangle - A(\theta)\}.$$

Note that the parameter vector θ and the vector of sufficient statistics $\phi(\mathbf{X})$ have the same length d . The term $A(\theta)$ is called the *log partition function*,

$$A(\theta) := \log \int_{\mathcal{X}^m} \exp\{\langle \theta, \phi(\mathbf{x}) \rangle\} \nu(d\mathbf{x}),$$

which is defined with respect to a reference measure ν such that $\mathbb{P}[\mathbf{X} \in S] = \int_S p_{\theta}(\mathbf{x}) \nu(d\mathbf{x})$ for any measurable set S .

The model smoothly assigns probabilities to all possible combinations. Graphical models cover all data, even those that are rare. In contrast to frequent set mining, no data whose frequency is below some threshold are ignored.

2.2 Modeling Temporal Dynamics

MRF do not yet model the temporal dynamics of random variables over a certain time period of length T . Turning them into a spatio-temporal model needs some enhancement. In particular, a sequence of snapshots of a spatial model needs to be defined. For many phenomena, as are seasons, weeks, etc., there is also a period to be modeled. Hence, the new spatio-temporal model should compose probabilities from series of spatial models and observe periodic phenomena.

For a formal description, we define the *snapshot graph* at time t by $G_t = (V_t, E_t)$ for $t = 1, 2, \dots, T$. We create

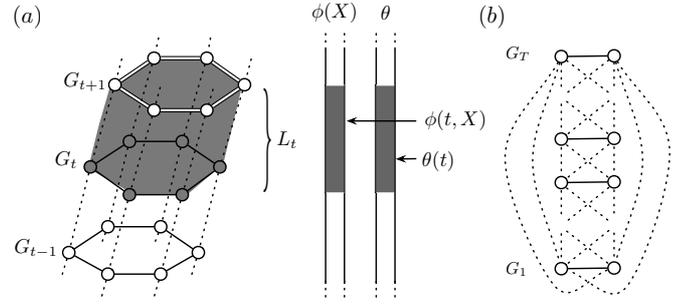


Figure 1: The spatio-temporal model consists of multiple snapshot graphs. The spatial and temporal edges are represented by solid and dotted lines, respectively. (a) A layer L_t (with simplified temporal edges) and its corresponding subvectors $\phi(t, \mathbf{X})$ and $\theta(t)$. (b) Detailed temporal dependency structure with periodic connections between top and bottom snapshot graphs.

each snapshot graph G_t by replicating a given *protograph* $G_0 = (V_0, E_0)$. The protograph represents a physical deployment of sensors, encoding the “spatial” structure of random variables that does not change over time. We also define the set of “temporal” edges $E_{t-1,t} \subset V_{t-1} \times V_t$ for $t = 2, \dots, T$, to represent dependencies between adjacent snapshot graphs G_{t-1} and G_t , $t = 2, \dots, T$: we assume 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 timeframes t and $t + 1$ can be chosen arbitrarily by users.

Periodicity is common to all random variables (given by T) over time and can be given by the user. That is, we map

$$\begin{aligned}G_t &:= G_{\text{mod}(t-1, T)+1}, \\ E_{t-1,t} &:= E_{\text{mod}(t-2, T)+1, \text{mod}(t-1, T)+1},\end{aligned}$$

for $t > T$, where $\text{mod}(a, b)$ is the remainder in scalar division of a by b . Finally, we connect the first and the last snapshot graphs, so that $E_{0,1} = E_{T,1} \subset V_T \times V_1$. The entire graph, denoted by G , consists of the snapshot graphs G_t stacked in order for timeframes $t = 1, 2, \dots, T$ 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 structure of G is shown in Figure 1.

We define a *layer* L_t as the subgraph of G containing all vertices of V_t and all edges of $E_{t,t+1}$, for $t = 1, 2, \dots, T$. For instance, a layer L_t is depicted with gray color and black nodes in Figure 1 (a). We define the subvectors of $\phi(\mathbf{X})$ and θ that correspond to a layer L_t as

$$\begin{aligned}\phi(t, \mathbf{X}) &:= (\phi_{v,a}, \phi_{vw,ab} \mid v \in L_t, (v, w) \in L_t, \forall a, b \in \mathcal{X}), \\ \theta(t) &:= (\theta_{v,a}, \theta_{vw,ab} \mid v \in L_t, (v, w) \in L_t, \forall a, b \in \mathcal{X}),\end{aligned}$$

which yields a functional form for the sufficient statistics and the parameters. By construction, the layers L_1, L_2, \dots, L_T define a nonoverlapping partitioning of a graph G , which allows us writing

$$\langle \phi(\mathbf{X}), \theta \rangle = \sum_{t=1}^T \langle \phi(t, \mathbf{X}), \theta(t) \rangle.$$

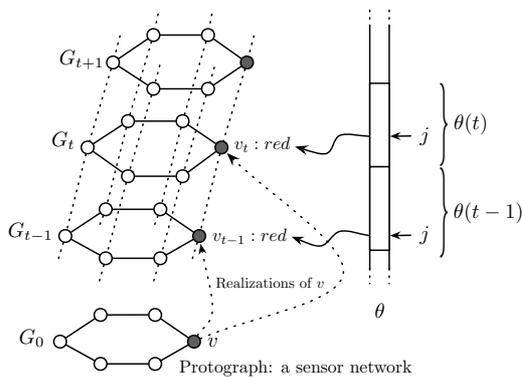


Figure 2: An example of indexing for a node:state pair over time. A sensor modeled by the node v in the protograph G_0 has its “realizations” v_{t-1} and v_t at timeframes $t-1$ and t respectively. Then the pairs $v_{t-1,red}$ and $v_{t,red}$ are located at the same index j in the subvectors $\theta(t-1)$ and $\theta(t)$, resp. We also call the index j is “associated” with the sensor v .

The subvectors $\phi(t, \mathbf{X})$ and $\theta(t)$ have the same length d/T for all $t = 1, 2, \dots, T$: we refer to as $d' := d/T$. Note that the subvectors should be “aligned”, in the sense that the j -th elements in all subvectors must point to the same sensor(s) over time. We illustrate this in Figure 2.

It is easy to adjust the model to a certain application. The user can indicate, how much time a period should take. She can also indicated the granularity of the model, i.e. the time gap between layers. For instance, if the period is a day and the gap between snapshots is 10 minutes, the model consists of 144 layers, since $144 \cdot 10 = 1440$ which is the number of minutes per day. In principle, the temporal distance between two measurements might vary over the network, but using equidistant measurements fits the behavior of most sensors.

2.3 Training and Prediction

The parameters $\theta \in \mathbb{R}^d$ are obtained by Maximum Likelihood Estimation, whereby the Likelihood of a particular θ given some data set \mathcal{T} is defined as

$$\mathcal{L}(\theta|\mathcal{T}) := \prod_{i=1}^N p_{\theta}(\mathbf{x}^{(i)}). \quad (3)$$

Here, \mathcal{T} is a set of $N = |\mathcal{T}|$ historical sensor readings and $\mathbf{x}^{(i)}$ is the i -th joint reading of all sensors in the network. If our model is implemented directly into a sensor, we do not want it to store its complete history of measurements. Therefore, we take the logarithm of the Likelihood (3) and rearrange, such that it only depends on the average value or the *empirical expectation* $\mathbb{E}[\phi(\mathbf{x})]$ of our sufficient statistic. Note that the extra $\frac{1}{N}$ -factor does not change the optimal solution.

$$\ell(\theta|\mathcal{T}) := \frac{1}{N} \sum_{i=1}^N \log p_{\theta}(\mathbf{x}^{(i)}) = \langle \theta, \mathbb{E}[\phi(\mathbf{x})] \rangle - A(\theta) \quad (4)$$

This implies that each sensor has to store an aggregate of historical readings instead of a full history. These aggregates

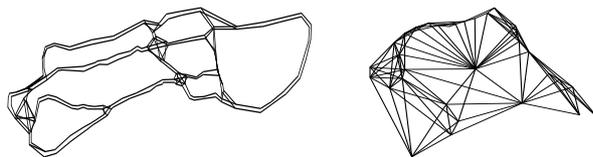


Figure 3: Spatial structures used in our examples. Left: Partial highway network of North Rhine-Westphalia, Germany. Right: Bouy network in the Pacific Ocean measuring the sea level.

are the empirical marginal probabilities presented in (2). Taking derivatives of (4) it follows that

$$\frac{\partial \ell(\theta|\mathcal{T})}{\partial \theta_{v,a}} = \tilde{\mathbb{E}}[\phi_{v,a}(\mathbf{x})] - \hat{\mathbb{E}}[\phi_{v,a}(\mathbf{x})],$$

whereby the *estimated expectation* $\hat{\mathbb{E}}[\phi_{v,a}(\mathbf{x})]$ is computed by Belief Propagation (BP) [16]. BP is also known as Sum-Product Algorithm in the context of factor graphs [8]. The objective $\max_{\theta} \ell(\theta|\mathcal{T})$ can then be solved by any first-order optimization method. We ran gradient descent optimization with an elastic-net regularization [24] until convergence to find proper model parameters.

The prediction (query) is executed by the BP algorithm. For this purpose, we fix the measurements of some given sensors, say u at time $t-1$ and w at time $t-h$, and run BP until convergence. The computed marginal probabilities for any sensor v_t will approximate $p_{\theta}(X_{v_t} = a | x_{u_{t-1}}, x_{w_{t-h}})$. Thus, the predicted value of each sensor v at time t is

$$x_{v_t}^* = \arg \max_{a \in \mathcal{X}} p_{\theta}(X_{v_t} = a | x_{u_{t-1}}, x_{w_{t-h}}). \quad (5)$$

We will use this for our exemplary predictive analysis of sensor networks in the following section.

In case of continuous $t \in \mathbb{R}$, a prediction can still be made by interpolating the probability between the two surrounding layers $[t]$ and $[t]$ and choosing its maximum point. A first-order interpolation is given by

$$\tilde{p}_{\theta}(X_{v_t}) = p_{\theta}(X_{v_{[t]}}) + (p_{\theta}(X_{v_{[t+1]}}) - p_{\theta}(X_{v_{[t]}}))(t - [t]).$$

3. APPLICATIONS

Graphical models produce probabilistic answers for queries. This gives a good picture of situations as is needed for decision makers. First experiments with traffic and tsunami data show that, indeed, the new spatio-temporal model can answer questions as those listed in the introduction. This is promising for its use in contexts of monitoring and disaster management.

We use two real-world data sets for evaluation, where each data set consists of a spatial protograph $G_0 = (V_0, E_0)$ with a set of sensors V_0 and connections E_0 and a set \mathcal{T} of historical sensor readings. Our open source implementation of spatio-temporal MRF is available for download (<http://sfb876.tu-dortmund.de/stmrf>).

3.1 Data Sets

Traffic. The first set comes from the traffic situation data on German highways, available at the Online Traffic Information System (<http://autobahn.nrw.de>). We take the measurements for highways in the state of North Rhine-Westphalia of Germany, consisting of the number of vehicles and their average speed per minute, and the occupancy rate of the highway region covered by each sensor. The locations of the corresponding sensors are shown in Figure 4(a). We use the data from July to October in 2010 for training. In total, the data set contains more than 200 million sensor readings.

To prepare a protograph, we first split highways into segments and create a node for the segments with high population around, since we want to make predictions on such segments. The rest segments are considered to be edges connecting the nodes. If there are more than one sensor in a node, we use an average of sensor readings as a measurement for the node. The measurements are discretized into the four states $\mathcal{X} = \{\text{free, low congestion, medium congestion, high congestion}\}$ as suggested by the german *Bundesanstalt für Straßenwesen* (BASt) for this kind of measurements [3]. After removing completely faulty sensors, our protograph contained 186 nodes and 237 edges.

Sea-Level. We use the sea level measurements from the Deep-ocean Assessment and Reporting Tsunamis (DART) buoys on the Pacific Ocean (available at <http://ngdc.noaa.gov/hazard/recenttsunamis.shtml>). Among the seven recent tsunamis with available data, we use six, discarding the “Solomon Islands” tsunami (August 1st, 2007) due to too many faulty measurements. A protograph of selected DART buoys is created based on a nearest neighbor graph, where distances between buoys are determined by their locations on the Pacific: the graph contained 33 nodes and 114 edges.

We take the measurements of each tsunami for four days, so that each tsunami event will locate on the second day. We set the sampling rate to be one per every 15 minutes, since it was the majority value among sensors, taking averages whenever we have finer resolution data around a sampling point. We normalized the data, discretizing them into ten equal-sized bins. This results in $\mathcal{X} = \{1, 2, \dots, 10\}$ for each sensor. The data set consists of 1.4 million sensor readings in total and all of them were used for training since each sensor is covered by only few tsunamis.

3.2 Modelling Traffic Networks

The Tihange Nuclear Power Station (NPS) is one of the two large-scale nuclear power plants in Belgium. It is located on the right bank of the Meuse River in the Belgian district of Tihange, part of Huy municipality in the Walloon province of Liège. The plant has three pressurized water reactors, with a total capacity of 2985 megawatts and makes 52 percent of the total Belgian nuclear generating capacity. The plant began operation in 1975. The distance between Tihange NPS and the nearest sensor in the German Ruhr area amounts to 88.8 miles.

One immediately sees that the street network is dense (w.r.t. average node degree) in and near the Ruhr area and is rather

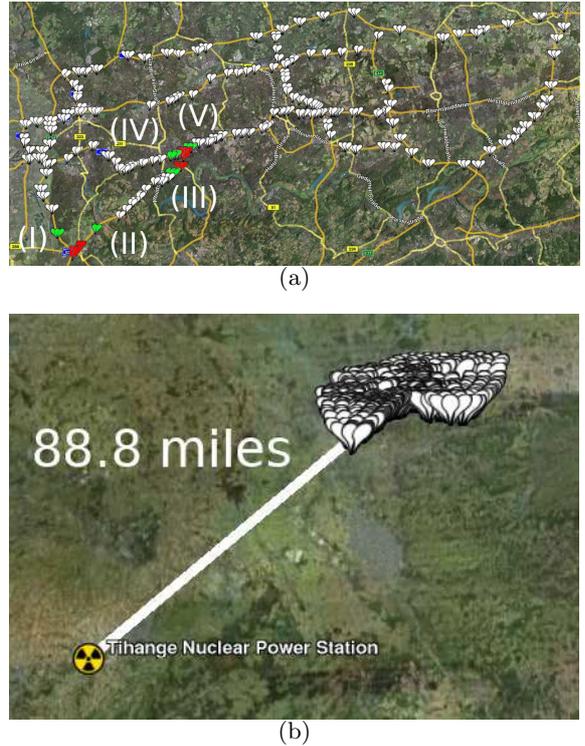


Figure 4: Image (a) show the sensor network which generated the traffic data set. Each flag indicates a single sensor. The second image (b) shows the beeline between the traffic network and the belgium nuclear power station Tihange.

sparse outside. In case of a nuclear disaster in Tihange, it is reasonable to expect a large amount of citizens to flee in north-east direction. Hence, heavy traffic at the south-west corner of the traffic network is to be expected. If there, the heavy traffic arrives at time t , what will the traffic be at time $t + h$ at other points? The spatio-temporal model is trained on the regular traffic given in the traffic data. The assumption is, that the probabilities as learned from these data, do express the dynamics of the network. The correlations of all nodes over all times are characterized. Hence, the model can answer the probability of heavy traffic at *any* point and for *any* h , given heavy traffic in the south-west Ruhr area. Inspecting this helps to decide, e.g., where to open areas for traffic. In this example, we built a small model with 12 layers, each covering a timeframe of 2 hours. Our query asks which way the majority of the refugees will take:

“Given the highest congestion level at the red sensors (Figure 4(a)) at 6 o’clock, how will the expected congestion level at the green sensors change?”

Some results are shown in Table 1. The model generates several tables indicating the probabilities for all the different states of all sensors in the network as well as joint probabilities of sensor pairs. For illustration, the estimated probabilities are shown for the sensors (I) to (V) which are

marked green in Figure 4(a). The first column lists the time. The second column contains the prior probabilities without any given event or observation. The third column contains the conditional probabilities given induced measurements of high congestion at the red sensors (Figure 4(a)) at 6 o'clock.

Some sensors are more sensitive to the highly congested regions than others. In this setup, only sensor (III) has a lower conditional probability than prior probability at 8 o'clock, directly after the induction. This is actually what one could expect, because sensor (III) measures traffic back into the south-west direction, which is the origin of our hypothetical disaster. Note, that this behavior is extracted from the training data without selecting a special subset of measurements. It is just the model as trained from regular measurements, once given heavy traffic at 6 o'clock (column -post) and once for the usual distribution of states at 6 o'clock (column -pre) on the red sensors. This shows that even very rare events are captured correctly by the generative model.

3.3 Modelling Buoy Networks

Recent Tsunami events shows the demand for predictive analysis of sea-level data. We illustrate how answers will look like in a hypothetical tsunami situation similar to the one in Section 1, e.g.

“Given a high sea level near Tokyo, what will be level near San Francisco after 3, 6, 9, and 12 hours?”

To simulate the situation, we pick two DART stations “21418” near Tokyo and “46411” near San Francisco in our Tsunami sensor graph (Figure 5). Our spatio-temporal model for the network has 384 snapshot graphs corresponding to sampling points at every 15 minutes. The training data covers four days by 384 snapshot graphs and a tsunami event occurs the second day. The state of station “21418” was fixed to “9”, which stands for “maximally high” at the 97-th snapshot graph (beginning of the second day). Then we make predictions for the station “46411” at the 109th (+3 hours), 121st (+6 hours), 133rd (+9 hours), and 145th (+12 hours) snapshot. The results are summarized in Figure 6. We can see the distribution for +9 and +12 after the event is shifted toward right, i.e. high sea level (the plot on the right), compared to the distribution of a normal situation (on the left). This result matches well with the Tsunami wave propagation prediction on http://ngdc.noaa.gov/hazard/dart/2011honshu_dart.html, which tells the wave arrival will be between +10 and +11 hours after the event.

4. CONCLUSIONS

There are several applications where data mining needs to take into account spatial as well as temporal information. If such a spatio-temporal model is probabilistic, it allows a prediction for any point in time and any location. In this paper, we have presented such a model. It is built on top of graphical models. We described training, prediction and interpolation of our models. Two example applications are designed for a better planning of processes or better disaster management. The spatio-temporal model gives probabilistic answers to complex queries which might become critical

	(I)-pre	(I)-post
6:00 am	0.0304424	0.0407377
8:00 am	0.0162724	0.0209663
10:00 am	0.0103336	0.00900889
12:00	0.00759318	0.00625894
14:00 pm	0.00636271	0.00523515
16:00 pm	0.00586461	0.00484635
18:00 pm	0.0059501	0.00496732
20:00 pm	0.00645001	0.00542935
22:00 pm	0.0100356	0.00861825
	(II)-pre	(II)-post
6:00 am	0.0214906	0.0351695
8:00 am	0.009977	0.0148234
10:00 am	0.00583667	0.00509503
12:00	0.004288	0.00353086
14:00 pm	0.00360725	0.00294739
16:00 pm	0.00340079	0.00278669
18:00 pm	0.00342298	0.00283451
20:00 pm	0.00362761	0.00306696
22:00 pm	0.00536169	0.00473857
	(III)-pre	(III)-post
6:00 am	0.0451546	0.0445284
8:00 am	0.0242879	0.0223179
10:00 am	0.0157111	0.0131034
12:00	0.0125312	0.0102323
14:00 pm	0.0120291	0.00979062
16:00 pm	0.0135301	0.0111008
18:00 pm	0.0172476	0.0147254
20:00 pm	0.022562	0.02058
22:00 pm	0.0293905	0.0278761
	(IV)-pre	(IV)-post
6:00 am	0.01818	0.0277054
8:00 am	0.00876866	0.0112617
10:00 am	0.00572315	0.00409307
12:00	0.0047374	0.00322085
14:00 pm	0.00465079	0.00319654
16:00 pm	0.00512045	0.00357861
18:00 pm	0.0061346	0.00438186
20:00 pm	0.00848675	0.00641204
22:00 pm	0.0165911	0.0138931
	(V)-pre	(V)-post
6:00 am	0.0144675	0.02355
8:00 am	0.00718417	0.0108755
10:00 am	0.00455275	0.00356956
12:00	0.00345458	0.00255771
14:00 pm	0.00300312	0.00222569
16:00 pm	0.00292774	0.00218345
18:00 pm	0.00322762	0.00245241
20:00 pm	0.00405396	0.00319741
22:00 pm	0.00796892	0.00665439

Table 1: This table shows probabilities for the high congestion state of sensors (I) to (V) over time. The “-pre” column shows prior probabilities for being in high congestion state without any given event and the “-post” column shows the conditional probabilities given that the event *high congestion from south-west* occurs at 6:00 am.



Figure 5: Buoy network on the Pacific Ocean. Each diamond represents a single buoy. Buoy “21418” near Tokyo is highlighted in red and buoy “46411” near San Francisco in green.

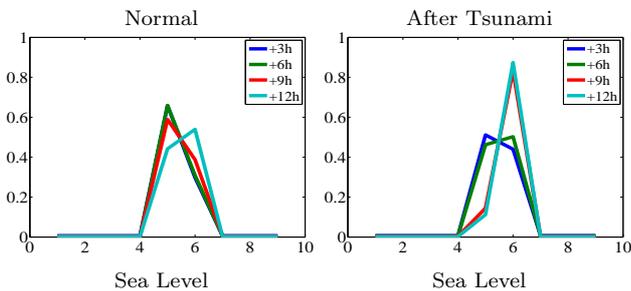


Figure 6: Probabilistic answers to a hypothetical Tsunami query, “What will happen near San Francisco +3, +6, +9 and +12 hours after a Tsunami has been detected near Tokyo?” The left plot is the prediction on sea level near San Francisco without the event, and the right is after the event.

in disaster scenarios. The first example was an hypothetical nuclear disaster in a Nuclear Power Station in Belgium. The effect of fleeing people is to be predicted for managing the traffic in the Ruhr area. The model is trained on regular behavior of traffic in the highway network of the Ruhr area. The model is then applied with a high volume traffic at the first local node where the fleeing people from Belgium would enter the Ruhr area. The results show that even very rare events with low probability might give some insight into the interaction of measured entities. This is in contrast to frequent set based approaches which delete events with a probability lower than a threshold. Frequent set based methods would deliver patterns including heavy traffic at a certain node, e.g., the south-west corner of the Ruhr area. A user may inspect what other states co-occur in these patterns. Depending on the threshold for minimal support several interesting correlations are excluded. Moreover, we cannot answer any question about what would happen, if there were an unusual traffic jam, because the patterns are found from the training set and fixed. The patterns do not model the situation, but only characterize the situations that have been perceived. Emergency cases are not given by the data. In contrast, probabilistic modeling the traffic allows to propagate assumed, not perceived states to the network.

In contrast to other existing spatio-temporal graphical model approaches, e.g. [23] or [7], our approach is generic in the sense that we do not make any assumptions on the spatial-structure of a model. Although we used a temporal first-order dependency between layers in this paper, the user of our model is free to choose a different order.

The second example dealt with sea-level prediction in the Pacific Ocean which might become an important prerequisite for disaster management in tsunami scenarios. It illustrates that the once trained model answers questions for all points in time and all locations. The output can graphically be presented as curves of probabilities. The example picture showed that after 9 hours, the likelihood of a sea level of 6 m or higher at San Francisco is more than 80%, if an extremely high sea level has occurred at Tokyo. It is not necessary that this high level at Tokyo has ever been measured. The trained model is applied to the high sea level value at Tokyo and delivers probabilities for all sea levels at all nodes.

For future work, it is planned to enhance our method by stream/real-time processing. This will allow evaluations where our model is implemented directly into the sensor network. Another possible extension is the processing of heterogeneous sensor data. Actually, the model does not require homogeneity of the nodes, but its current implementation uses a common scheme of attributes for all nodes.

5. 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.

6. REFERENCES

- [1] A. Camerra, T. Palpanas, J. Shieh, and E. J. Keogh. isax 2.0: Indexing and mining one billion time series. In G. I. Webb, B. Liu, C. Zhang, D. Gunopulos, and X. Wu, editors, *Procs. of IEEE ICDM*, pages 58–67, 2010.
- [2] D. Campbell. Is it still Big Data if it fits in my pocket? In *Procs. VLDB 2011*, 2011.
- [3] B. für Straßenwesen. Merkblatt für die Ausstattung von Verkehrsrechnerzentralen und Unterzentralen (MARZ). 1999.
- [4] A. R. Ganguly and K. Steinhaeuser. Data mining for climate change and impacts. In *Procs. of IEEE Int. Conf. Data Mining (ICDM) Workshops*, pages 385 – 394, 2009.
- [5] S. Günemann, H. Kremer, C. Lauffkötter, and T. Seidl. Tracing evolving subspace clusters in temporal climate data. *Data Mining and Knowledge Discovery*, 24(2):387–410, 2012.
- [6] J. M. Hammersley and P. Clifford. Markov fields on finite graphs and lattices. *Unpublished*, 1971.
- [7] R. Huang, V. Pavlovic, and D. 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 - MLVMA '08*, Marseille, France, 2008.

- [8] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. on Infor. Theory*, 47(2):498–519, 2001.
- [9] Z. Li, J. Han, B. Ding, and R. Kays. Mining periodic behaviors of object movements for animal and biological sustainability studies. *Data Mining and Knowledge Discovery*, 2011.
- [10] M. Lippi, M. Bertini, and P. Frasconi. Collective traffic forecasting. In *Procs. of ECML PKDD (Part 2)*, pages 259–273, 2010.
- [11] Y. Low, J. Gonzalez, A. Kyrola, D. Bickson, C. Guestrin, and J. M. Hellerstein. Graphlab: A new parallel framework for machine learning. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, Catalina Island, California, July 2010.
- [12] D. Luckham. *The Power of Events - An Introduction to Complex Event Processing in Distributed Enterprise Systems*. Addison Wesley, 2002.
- [13] S. F. Marinossou, R. Chrobok, A. Pottmeier, J. Wahle, and M. Schreckenber. Simulation Framework for the Autobahn Traffic in North Rhine-Westphalia. In *Cellular Automata - 5th Int. Conf. on Cellular Automata for Research and Industry, ACRI*, pages 2977–2980. Springer, 2002.
- [14] M. May and L. Saitta, editors. *Ubiquitous Knowledge Discovery*, volume 6202 of *Lecture Notes in Artificial Intelligence*. Springer, 2010.
- [15] K. Morik, K. Bhaduri, and H. Kargupta. Introduction to data mining for sustainability. *Data Mining and Knowledge Discovery*, 24(2):311–324, 2012.
- [16] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.
- [17] T. Rakthanmanon, E. J. Keogh, S. Lonardi, and S. Evans. Time series epenthesis: Clustering time series streams requires ignoring some data. In D. J. Cook, J. Pei, W. Wang, O. R. Zaïane, and X. Wu, editors, *Procs. of IEEE ICDM*, pages 547–556, 2011.
- [18] G. Sagy, D. Keren, I. Sharfman, and A. Schuster. Distributed threshold querying of general functions by a difference of monotonic representation. In *Procs. of the VLDB Endowment*, volume 4, 2011.
- [19] M. Steinbach, P.-N. Tan, V. Kumar, S. A. Klooster, and C. Potter. Discovery of climate indices using clustering. In L. Getoor, T. E. Senator, P. Domingos, and C. Faloutsos, editors, *Procs. KDD*, pages 446–455. ACM, 2003.
- [20] D. Wang, E. A. Rundensteiner, and R. T. Ellison. Active complex event processing of event streams. In *Procs. of the VLDB Endowment*, volume 4, 2011.
- [21] J. Whittaker, S. Garside, and K. Lindveld. Tracking and predicting a network traffic process. *International Journal of Forecasting*, 13(1):51–61, March 1997.
- [22] R. Wolff, K. Badhuri, and H. 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.
- [23] Z. Yin and R. Collins. Belief Propagation in a 3D Spatio-temporal MRF for Moving Object Detection. *IEEE Computer Vision and Pattern Recognition (CVPR)*, June 2007.
- [24] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society B*, 67:301–320, 2005.