Spatiotemporal Anomaly Detection in Gas Monitoring Sensor Networks

X. Rosalind Wang[1,⋆], Joseph T. Lizier[1,2], Oliver Obst[1], Mikhail Prokopenko[1], and Peter Wang[1]

[1] CSIRO ICT Centre, Locked Bag 17, North Ryde, NSW 1670, Australia
Rosalind.Wang@csiro.au
[2] School of Information Technologies, The University of Sydney, NSW 2006, Australia

Abstract. In this paper[1], we use Bayesian Networks as a means for unsupervised learning and anomaly (event) detection in gas monitoring sensor networks for underground coal mines. We show that the Bayesian Network model can learn cyclical baselines for gas concentrations, thus reducing false alarms usually caused by flatline thresholds. Further, we show that the system can learn dependencies between changes of concentration in different gases and at multiple locations. We define and identify new types of events that can occur in a sensor network. In particular, we analyse joint events in a group of sensors based on learning the Bayesian model of the system, contrasting these events with merely aggregating single events. We demonstrate that anomalous events in individual gas data might be explained if considered jointly with the changes in other gases. Vice versa, a network-wide spatiotemporal anomaly may be detected even if individual sensor readings were within their thresholds. The presented Bayesian approach to spatiotemporal anomaly detection is applicable to a wide range of sensor networks.

1 Introduction

Since the 1980s, electronic gas monitoring sensor networks have been introduced in the underground coal mining industry. However, no current system can provide site specific anomaly detection. This means monitoring systems often give false alarms, which can be costly to the mining operation. The periodic variation in the gas concentration also increases the number of false alarms in these flat line threshold based systems. Further, current systems ignore the spatial relations between data gathered at different sensor network nodes. These spatial relationships between data could identify anomalies missed by individual sensors. Conversely, the spatial relationships could explain away the anomalies identified by the individual gas sensors, thus avoiding false alarms.

Currently, the existing system integrates and interprets incoming data in accordance with a pre-determined set of rules, produces a risk profile, and autonomously initiates a response to a breach of these rules. A problem with this approach is that no clear-cut definitions of abnormal situations with respect to the concentration of different gases exist, so that it is difficult to produce a good set of rules.

⋆ Corresponding author.

[1] The authors list after the lead author is in alphabetical order.

R. Verdone (Ed.): EWSN 2008, LNCS 4913, pp. 90–105, 2008.

The underground coal mining industry has been struggling with the issues of site-based moving threshold levels for critical gases since the introduction of electronic gas monitoring systems in the 1980s. No satisfactory, scientifically validated methodology is in existence that can provide a mine with its own specific moving threshold levels. Best guess estimates, universal rules-of-thumb and experience-based trigger points are the industry norm [1]. In this paper, we used Bayesian Networks as a means for unsupervised learning of temporal and spatiotemporal patterns in underground coal mining gas data, and applied the approach to spatiotemporal anomaly detection.

Section 2 presents the problem of anomaly detection in general sensor networks. In Sec. 3, we define the problem specifically for underground coal mining sensor networks. Section 4 describes the approach we took to learn and analyse the data. The results of identified anomalies are shown in Sec. 5. Finally, Sec. 6 presents the conclusions and future work.

2 Background

In order to be successful, sensor networks must detect, evaluate and diagnose patterns in diverse situations, forecast likely future scenarios, make decisions, initiate actions based on these decisions, and adapt to change. Adaptive anomaly detection in spatiotemporal sensor network data is, thus, one of the main challenges in this field. Conventional control theory and SCADA (Supervision Control And Data Acquisition) systems are employed for anomaly detection in these sensor networks, however, they are inadequate to deal with scenarios which require flexible acquisition and distribution of information.

For our particular application, each node in the sensor network monitors several different kinds of gases in order to ensure safety and productivity in a coal mine. We consider an existing system which takes measurements and interprets incoming data. The single nodes in the sensor networks cover wide areas. Since they are used for the prevention of hazards, rather than for recovery after a hazard, the position of each node is fixed and known. Our scenario also allows for the use of non-wireless communication between single nodes, whereas for applications in hazard recovery, cable-based communication could possibly be disrupted by collapsed roofs or explosions.

The application of sensor networks in coal mines seems to be natural, because several different kinds of data have to be collected for safety reasons. For example, in [2], a sensor network is used to detect leakages of gas, dust or water, and to monitor the density of oxygen in different areas. For this particular application, data from different nodes is used to create a qualitative overview, describing for example the extent of water leakages or areas with a high density of oxygen. For our application, monitoring gases at each node separately from each other is not sufficient to detect anomalies: densities of single gases at one location might appear normal, but the simultaneous measurement of densities of other gases at other locations could in fact indicate a potentially dangerous situation. Other properties of the scenario make the detection of abnormal situations more difficult: as mentioned above, there is no good definition of an abnormal situation, and one of the reasons for this is the rarity of abnormal events in the available data. Moreover, not only does the spatial distribution of gases have to be considered, but so too does the development of gas distributions over time.

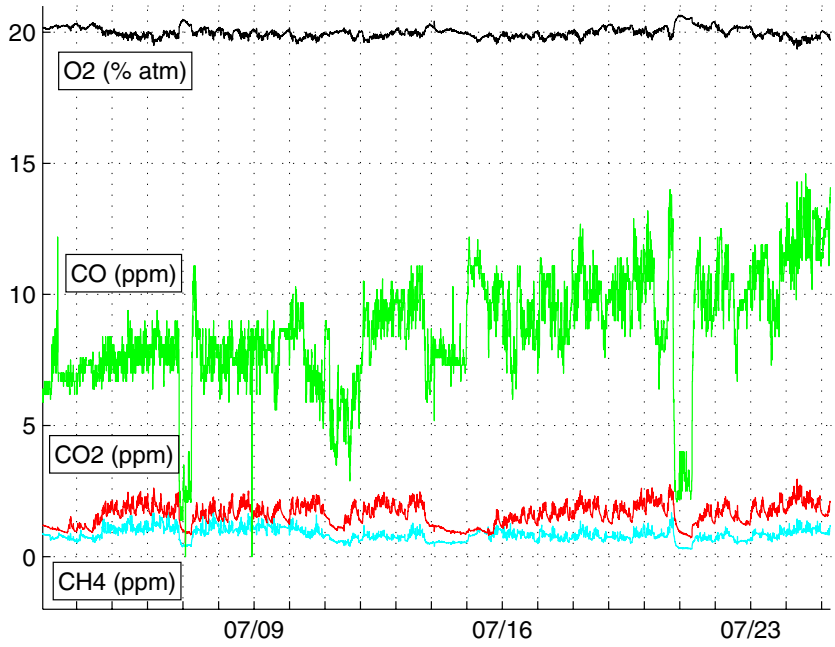


Fig. 1. Gas concentration data from a sensor node in an Australian coal mine. Data gathered for oxygen is in units of percentage of air, the other gases are in of parts per million. The horizontal axis indicates the time (mm/dd) the data was taken.

For the research presented in this paper, we are using data gathered from deployed sensor networks in existing Australian coal mines for testing the algorithms. Each sensor node measures gas concentration, e.g. at 30 second intervals, of a number of gases, e.g. methane (CH_4), carbon dioxide (CO_2), carbon monoxide (CO) and oxygen (O_2). Figure 1 shows the data from a sensor node in the first mine for three weeks in July 2006.

Intuitively, anomalies in our data are irregular patterns in multiple time series, e.g. a combination of $CH_4 - CO_2 - CO - O_2$. In general, the problem is to detect an abnormal event distributed over different sensors, although it is not clear what exactly "abnormal" means a priori. In our particular coal mine sensor network, all data is passed on to a central node, so that currently the problem of a distributed computation of abnormal situations is not pressing. Nevertheless, we pursue a method that has the potential to be distributively computed if required.

Methods to identify previously unseen, i.e. abnormal situations in data have previously been investigated [3,4]. The method in [3] uses self-organising maps (SOM) to describe normal system behavior, and to detect abnormal behavior. In order to use SOM, the authors present a new measurement to find out if a dataset and a map are matching based on a k-nearest neighbor approach.

The method introduced in [4] detects abnormal events in signals using Support Vector Machines (SVM). The method can be used online, i.e. without using a fixed training set: the last n observed input vectors are used for training. In the first of the proposed algorithms, a special kind of distance measure is used to compare the distance of a

new vector to a region created by the last n input vectors. A distance greater than a given threshold is considered as abnormal. The second algorithm is similar, but delays the output by a small number of measurements N. Then, the *set* of N new vectors is tested for abnormality in a similar way to the first algorithm, leading to a more robust approach.

Both approaches expect all the data to be present in a single node, i.e. they could be used in the centralised fashion that coal mine sensor network deals with the data currently. We have, however, chosen to use a method based on Bayesian networks, because it would support inference in both distributed and centralised settings (see also [5]). In addition, our approach directly computes a likelihood measure for new data, thus allowing unsupervised learning for anomaly detection.

3 Problem Definition

One of the major road blocks we face in anomaly detection inside underground coal mines is complete lack of ground truth. This is because every mine is unique, so what is considered to be an anomaly in one mine may not be an anomaly in another. Mining experts do not have general purpose rules for anomaly detection that are applicable to every site. Therefore, our purpose is to devise an adaptive system that learns from the data specific to a mine, and identifies anomalies that are specific to the mine.

3.1 Temporal Anomalies in a Single Gas

Many current automatic detection systems use a flat baseline or threshold for anomaly detection. However, gas concentration in mines have a moving baseline depending on factors such as atmospheric pressure. That is, the mine "breathes" through the day, and the concentration of the various gases increases and decreases periodically. A flat baseline system does not capture this characteristic of the data, thus giving many false alarms and false negatives.

We consider an anomalous event or simply **event** in the time series data as a data point that results in a low likelihood given a model we have constructed of the time series. That is, the resulting likelihood value of the data point is an outlier from the general distribution of likelihood values of the other data. We will define likelihood, outliers and consequently the term 'event' formally after presenting the approach to the problem in Sec. 4.

This problem can be easily seen in the CO data in Fig. 1. For example, the data from July 16th to July 20th show a cyclical pattern in the concentration. A flat baseline system might assume the peak around July 18th is an anomaly, while we can see it's just a part of the moving cycle. Figure 4 also illustrates cyclical patterns.

3.2 Joint Temporal Anomalies in Multiple Gases

When several single events occur at the same time, they indicate a higher importance event. The current literatures identifies these as composite events and group events. A **composite** event, as defined by Kumar *et al.* [6] is a combination of two or more

single events. Jiao *et al.* [7] used the term "group level event" in similar fashion to the composite event, that is the aggregation of multiple local events. Informally, a **group** event occurs when in a group of sensors, each sensor identifies an event.

While the aggregation of single events might be adequate for the situations described in the papers above, we need to define other types of events for our data. In Fig. 1 for example, we can see around July 7th, the concentrations of CO, CO_2 both dropped, however, at the same time the concentration of O_2 increased. These single events, as a combination, is considered safe by mining experts. Conversely, while no events may be identified by isolated analysis of single gases, as a combination, they may be considered an event. We define three new terms in event detection for sensor networks: joint, explained and implicit events. Below we describe each event informally, they will be defined mathematically in Sec. 4.4.

A **joint** event is a combination of data from sensors that results in a low likelihood given the model for the combination of single sensors. Consequently, we define explained and implicit events where there is a difference of opinion in joint event and single events.

An **explained** event is where there are detected anomalies in single gases, but the combination of time series do not result in a joint event. The CO-CO_2-O_2 event situation described above would be classified as an explained event. The opposite to an explained event is an **implicit** event. This is when isolated analysis of single gases do not cause any alarms, however, as a joint event, these measurements are significant enough to trigger an alarm.

3.3 Network-Wide Spatiotemporal Anomalies

The events described above relate to gases at a single spatial location, however, they also apply to data of different sensor nodes in the network. In situations involving different sensor nodes in the network, a composite event would involve two or more single events at different nodes, and a group event is one where every sensor node in the group identified an event [7,6].

A network-wide "explained" event is when a truck passes through the mine. The exhaustion gases may trigger alarms in individual gases as concentration will increase suddenly. However, this event should not be considered a network-wide anomaly in the data, as other nodes jointly explain it away. An example of an "implicit event" in the network is an increase of methane at one location of the mine, accompanied by an increase in oxygen at another location. Thus no joint event is identified at each sensor node, but a network-wide joint event could be identified for the combination of the sensor nodes.

4 Approach to the Problem

Our approach to the problem of anomaly detection is to use Bayesian Networks (BNs). The networks are constructed via a learning process from some training data. When new observations are made, we can use inference on the network to find a likelihood value of the network given the new observations. An anomaly is identified if the likelihood value is low.

4.1 Bayesian Networks

A Bayesian Network is a graphical model that takes a statistical approach to learning. Statistical learning uses probability distributions to model variables that represent the gathered data, thus taking into account the stochastic nature of real data. Graphical models expose the underlying relationship between probabilistic variables in a simple and clear form.

Specifically, Bayesian Networks are a form of acyclic directed graph (ADG) [8] in that if one variable of the network is dependent on another, then the reverse cannot be true. This relationship between two variables is represented in BNs by the direction of an arrow connecting the two. The variables of a BN are called *nodes* of a BN. The node with an arrow pointing to it is dependent on the node with the same arrow pointing away from it. The nodes connected by an arrow have a parent/child relationship, where the *child* node is dependent on its *parent* node. (See Fig. 2 for one of the network structures used in this paper.)

In a Bayesian network, each random variable is independent of its non-descendants in the graph given the state of its parents. This independence can be exploited to reduce the number of parameters needed to characterise the network. Thus it is possible to efficiently compute posterior probabilities given some evidence or observations. One set of probability parameters are encoded for each variable, in the form of the local conditional distribution given the variable's parent. Using the independence statements encoded in the network, the joint probability distribution is uniquely determined by these local conditional distributions [9,10]. We present the general form of this joint probability distribution in the following paragraphs.

We use capital letters such as X, Y for names of random variables, and lower cases x, y for values taken by these variables. A set of variables such as $\{X_1, X_2, X_3\}$ are written as $\mathbf{X}$, likewise, a set of values such as $\{x_1, x_2, x_3\}$ are written as $\mathbf{x}$. Thus, $\mathbf{x}$ are values taken by $\mathbf{X}$.

Let $P(\mathbf{U})$ be a joint probability distribution over $\mathbf{U} = \{X_1, \ldots, X_k\}$, where X_i is a random variable expressed by a node of the network. A Bayesian Network for $\mathbf{U}$ is a pair $B = \langle G, \Theta \rangle$. The first component, G, represents the graph structure of the network. G is an ADG whose nodes correspond to the random variables $X_1, \ldots, X_k$, and whose edges represent direct dependencies between the variables. The second component, Θ, represents the set of conditional probabilities that quantify the nodes of the network. It contains a set of parameters $\theta_{X_i|\Pi_{X_i}} = P_B(X_i|\Pi_{X_i})$ for each node X_i, where Π_{X_i} denotes the set of parents of X_i in G. A Bayesian Network B defines a unique joint probability distribution over $\mathbf{U}$ given by

$$P_B(\mathbf{U}) = \prod_{i=1}^{k} P_B(X_i|\Pi_{X_i}) = \prod_{i=1}^{k} \theta_{X_i|\Pi_{X_i}}. \tag{1}$$

In a Bayesian Network the learning process is to estimate the parameter set Θ as well as to find the structure of the network, G. The objective in the learning is to find a $B = \langle G, \Theta \rangle$ that "best describes" the probability distribution over the training data [11]. In this paper, however, we will not be learning the structure of the networks.

4.2 Network Structures for the Problem

Let a single variable time series be $\mathbf{X} = \{x_1, x_2, \ldots, x_N\}$, where N is the total number of data points in the series. We can embed this data in a d-dimensional phase space as follows [12]:

$$\mathbf{y}_t = \left(x_t, x_{t-\tau}, \ldots, x_{t-(d-1)\tau}\right), \tag{2}$$

where τ is the time delay, d is the embedding dimension, and $t = d, d+1, \ldots, N$. Henceforth, we set $\tau = 1$, thus Eqn. 2 becomes:

$$\mathbf{y}_t = \left(x_t, x_{t-1}, \ldots, x_{t-d+1}\right), \tag{3}$$

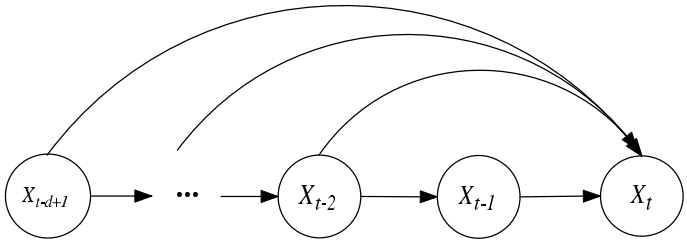


Fig. 2. Bayesian Network model used for learning and inference data in embedded phase space

Figure 2 shows the model used to learn this data. The network is constructed from the underlying dependencies in a time series, that is the data at time t is dependent on the data at time $t-1, \ldots, t-d+1$.[2] The joint distribution of the model is:

$$P(\mathbf{U}) = P(X_t|X_{t-1}, \ldots, X_{t-d+1})P(X_{t-d+1}) \prod_{k=1}^{d-2} P(X_{t-k}|X_{t-(k+1)}) \tag{4}$$

where $\mathbf{U} = \{X_{t-1}, \ldots, X_{t-d+1}\}$. All the nodes are modelled as one dimensional Gaussians. For example, a BN model of Fig. 2 with $d = 3$ has the dependencies as $X_{t-2} \rightarrow X_{t-1} \rightarrow X_t$ as well as $X_{t-2} \rightarrow X_t$. The joint distribution of the model will be $P(\mathbf{U}) = P(X_t|X_{t-1}, X_{t-2})P(X_{t-1}|X_{t-2})P(X_{t-2})$, where each $P(\cdot)$ is a Gaussian or a conditional Gaussian distribution.

Figure 3 shows a model that may be used to learn and inference the combination of three sensors in the system. In this case, the network is composed of three 'subnets', that is the sets of nodes $\{A_t, A_{t-1}, \ldots, A_{t-m}\}$, etc. Each subnet has the same network configuration as that of the network in Fig. 2. The value for m, that is, the length of the subnet, is not necessarily the value of d, which is the number of nodes for the network in Fig. 2. Since $\{B_{t-1}, \ldots, B_{t-m}\}$ is independent of A_i, and $\{C_{t-1}, \ldots, C_{t-m}\}$ is independent of A_t or B_t, we can write the joint distribution of the model as:

$$P(\mathbf{U}) \propto P(\mathbf{A})P(\mathbf{B})P(\mathbf{C})P(B_t|A_t)P(C_t|A_t)P(C_t|B_t), \tag{5}$$

where $P(\mathbf{A})$ is the joint distribution of $\{A_t, A_{t-1}, \ldots, A_{t-m}\}$, and similarly for $P(\mathbf{B})$ and $P(\mathbf{C})$.

[2] The network in Fig. 2 is simply a $d-1$-th order Markov model presented in the Bayesian Network representation.

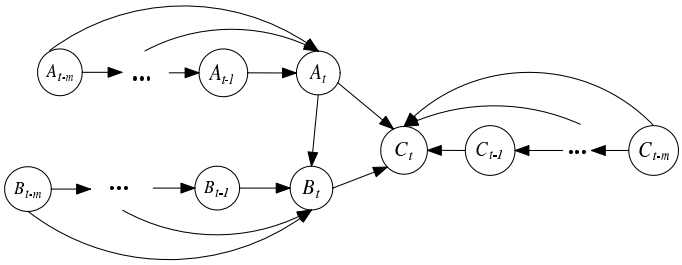


Fig. 3. Bayesian Network model used for learning the combined data from three sensors

Figure 3 describes a network for anomaly detection in the combination of gases at one spatial location, however, it can be easily adapted for detection across different spatial locations. For example, with the same network, A_t could be gas 1 from location 1, while B_t and C_t are gases 2 and 3 from location 2.

4.3 Learning and Inference

Since the structure of the network is known, only the parameter set Θ needs to be learnt. The Maximum Likelihood [13,14] algorithm is thus used to estimate Θ. In the ML estimator, the likelihood function, $p(\mathbf{x}|\theta)$, is treated as a function of θ for fixed $\mathbf{x}$, where x_j^t is the j-th data sample for the node X_t in the Bayesian Network. This *likelihood function* can be used to evaluate the choices of θ. The ML estimator chooses the value of θ that maximises the probability of the data $\mathbf{x}$:

$$\hat{\theta}_{ML} = \arg\max_{\theta} p(\mathbf{x}|\theta). \tag{6}$$

This learnt network can then be used to do inference on new data. That is, given the observed values of some of the nodes in the network, compute the probability distribution of the other nodes. Inference allows us to perform three types of analyses on the data:

1. *Prediction*, where the probability distribution of the child node can be computed given the values of the parents. In our case, the prediction of values in X_t in Figure 2 given the values of $\{X_{t-1}, \ldots, X_{t-d+1}\}$.
2. *Diagnosis*, where we can find the probability distribution of the parent node given the value of the child. In Figure 3 for example, given the values of B_t and C_t, we can find the values of A_t.
3. *Anomaly detection* using the likelihood values, which is actually a byproduct of inference operation. The likelihood value measures how well the observations fit the Bayesian Network model. Anomalies would result in a low likelihood value, while data that fit the model well will result in high likelihood values.

4.4 Anomaly Definitions

We will now define anomalies and the various events described in Sec. 3 formally. For all definitions below, the null hypothesis, H_0 is the hypothesis that the evidence is true,

and the p-value is the probability of how well the evidence supports the H_0 hypothesis (smaller p-values favour the rejection of H_0). The significance value, α is set such that if $p < \alpha$, then the null hypothesis is rejected.

Let $L_{\mathbf{y}_t} = L(\mathbf{y}_t|\theta_{B_\mathbf{y}})$ be the log likelihood of the Bayesian Network given data point $\mathbf{y}_t$ in a d-dimensional phase space (as shown in Figure 2), and $P_{L_\mathbf{Y}}(\theta_{L_\mathbf{Y}})$ be the distribution of $L_{\mathbf{y}_t}$ overall. Then

Definition 1. $\mathbf{y}_t$ *is an* ***event*** *iff the following* H_0 *is rejected:*

$H_0 : L_{\mathbf{y}_t} \sim P_{L_\mathbf{Y}}(\theta_{L_\mathbf{Y}})$

For the spatiotemporal events, let $\mathbf{u}_t = \{\mathbf{y}_t^1, \mathbf{y}_t^2, \ldots, \mathbf{y}_t^n\}$ be the set of n data points in the d-dimensional phase space at time t. For example, with three gases A, B, and C, $\mathbf{u}_t = \{\mathbf{a}_t, \mathbf{b}_t, \mathbf{c}_t\}$, where $\mathbf{a}_t = \{a_t, a_{t-1}, \ldots, a_{t-(d-1)}\}$, etc.

Definition 2. $\mathbf{u}_t$ *is a* ***composite event*** *when two or more of* $\{\mathbf{y}_t^1, \mathbf{y}_t^2, \ldots, \mathbf{y}_t^n\}$ *is an event.*

Definition 3. $\mathbf{u}_t$ *is a* ***group event*** *iff* $\mathbf{y}_t^i$ *is an event,* $\forall \mathbf{y}_t^i \in \mathbf{u}_t$.

In Definitions 1–3, we use log likelihood of the BN for data from a single sensor. Now we utilise the log likelihood of the BN for combined sensors. Let $L_{\mathbf{u}_t} = L(\mathbf{u}_t|\theta_{B_\mathbf{u}})$ be the log likelihood of the Bayesian Network for $\mathbf{u}_t$ (e.g. as shown in Figure 3), and $P_{L_\mathbf{U}}(\theta_{L_\mathbf{U}})$ be the distribution of $L_{\mathbf{U}_t}$.

Definition 4. $\mathbf{u}_t$ *is a* ***joint event*** *iff the following* H_0 *is rejected:*

$H_0 : L_{\mathbf{u}_t} \sim P_{L_\mathbf{U}}(\theta_{L_\mathbf{U}})$

Definition 5. $\mathbf{u}_t$ *is an* ***explained event*** *iff* $\mathbf{u}_t$ *is not a joint event but any one of* $\{\mathbf{y}_t^1, \mathbf{y}_t^2, \ldots, \mathbf{y}_t^n\}$ *is an event.*

Definition 6. $\mathbf{u}_t$ *is an* ***implicit event*** *iff* $\mathbf{u}_t$ *is a joint event but none of* $\{\mathbf{y}_t^1, \mathbf{y}_t^2, \ldots, \mathbf{y}_t^n\}$ *is an event.*

The possibility of explained and implicit events is due to the fact that in general, $L_{\mathbf{u}_t}$ and $\sum_{i=1}^{n} L_{\mathbf{y}_t^i}$ may differ significantly.

5 Results and Discussion

To learn the Bayesian Network using the phase space representation of Eqn. 2 we used $d = 20$ and $\tau = 1$. The process of finding d is described in detail in Appendix A. Fraser and Swinney suggested to use the mutual information method to find τ [15]. However, we found through experiments, that $\tau = 1$ gives much better inference results. We trained the networks using the first half of the data as presented in Fig. 1, and run inference on the second half of the data. Table 1 shows the normalised root mean square error (NRMSE) of the inference. NRMSE gives a useful scale-independent measure of error between data sets of different ranges.

Figure 4 illustrates a cyclical baseline for a 7 day period of CH_4 sensor data from a second mine in Australia, contrasting actual and predicted data. A cyclical baseline can

Table 1. Prediction errors (NRMSE) for the data using the Bayesian Network in Fig. 2

Methane	Carbon dioxide	Carbon monoxide	Oxygen
0.0404	0.0210	0.0468	0.0302

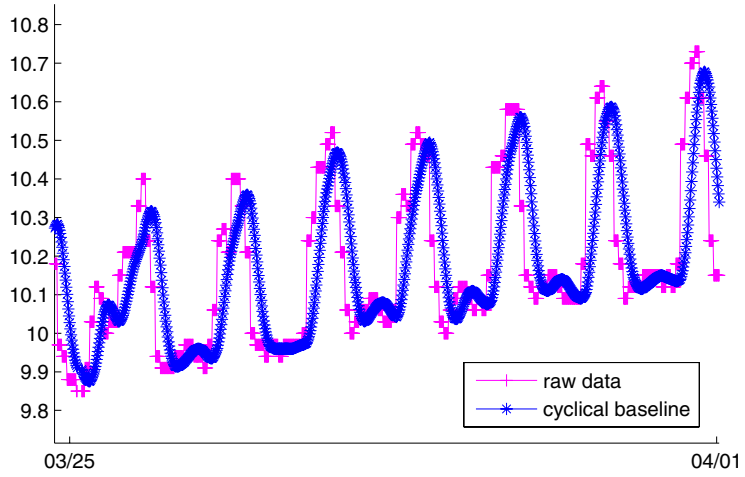


Fig. 4. Cyclical prediction of CH_4 for a 7 day period. Data are from a second mine in Australia.

be used to set moving thresholds around it, so that the fluctuating sensor data are within the thresholds. This in general will reduce false alarms.

Figure 5 shows the results of likelihood computation for the four gases in the data. For each sub-figure, we plot the actual data (bottom plot, left scale) with the logarithm of the likelihood (top plot, right scale). The Kolmogorov-Smirnov (KS) hypothesis test [16] was used to determine the anomalies from the likelihood values. The KS test is used because it can compare the test sample against any distribution, and it can be seen from Fig. 5 the log likelihoods do not fit a normal distribution, which is assumed by t-test or z-test. We applied the KS test using the extreme value distribution, which is a distribution skewed to the left as fitting for the likelihood results. The parameters of the distribution are set to be the mean and standard deviation of the likelihood values in each set of results.

It should be pointed out that in reality, there are no anomalies of real concern in this data set. That is because in an actual mining operation, events that are significant enough to raise an alarm and evacuate the mine are very rare. In most cases, any significant change in data would result in an immediate investigation of the situation and so the potential anomaly event would be avoided in the real data. To demonstrate the algorithm, we ran the KS test using $\alpha = 0.012$, that is the null hypothesis H_0 is rejected if $p < 0.012$. Normally, $\alpha = 0.01$ would be the first choice for anomaly detection through hypothesis tests [5,17], however with our data set, at $\alpha = 0.01$ no events were identified. The value we've chosen, $\alpha = 0.012$ allow us to demonstrate the flexibility of the method. The resulting events detected by the KS test are plotted as red dots in Fig. 5 above the likelihood values.

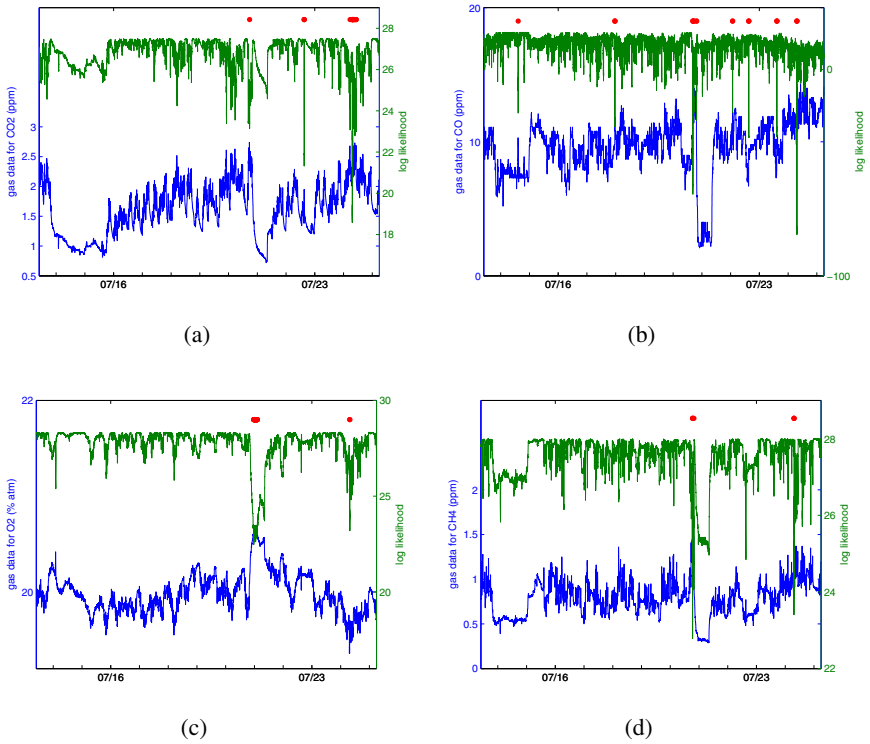
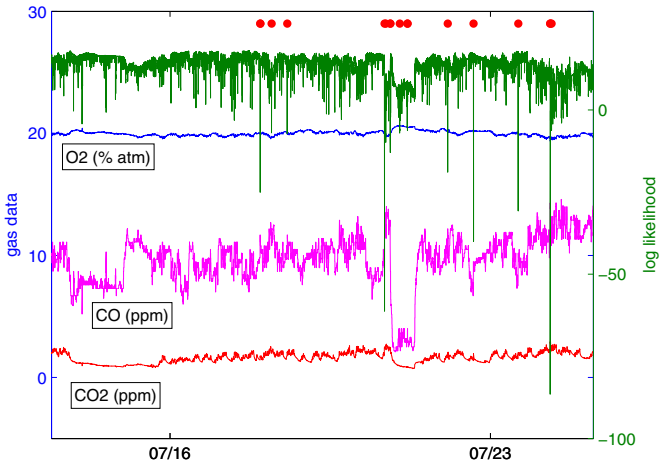


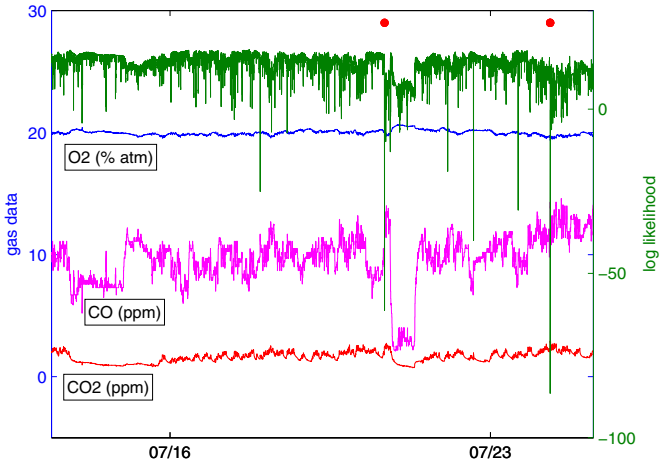
Fig. 5. Results of anomaly detection for single gases: (a) CO_2, (b) CO, (c) O_2, and (d) CH_4. For each plot, the bottom curve shows the data collected from the gas sensor, the top curve shows the log likelihood found given the data, and the dots at the top show the anomalies as determined by the algorithm.

Figure 5(a) shows there are three distinctive candidate anomaly events in the CO_2 data for this time period. Note the second anomaly identified by the system around July 23rd. This corresponds to a sudden jump in gas concentration during an interval where the gas concentration is decreasing. This highlights the advantage of using a system that has learnt from past events. This type of anomaly cannot be detected with a flat baseline benchmark, as at this particular time the gas concentration is lower than the two nearby peaks. Another interesting feature is that the large drop in gas concentration around July 21st was not identified as an anomaly, while a threshold system may do otherwise. Figure 1 showed that a similar event happened two weeks earlier around July 7th. However, the peak in gas concentration just before this dip was identified as an anomaly as this was an unusual event in the history of the data set.

Results of anomaly detections in CO, O_2 and CH_4 in Fig. 5(b)–(d) show similar anomalous and normal events as those of CO_2 results. Of particular interest is the last anomaly identified in the CO data, since at the scale presented, it is difficult to see why this particular region was identified as an anomaly with such a low log likelihood. However, upon closer inspection, we found that this is caused by a difference of 1.7 ppm between two consecutive data points. That is, in 30 seconds, the CO concentration



(a)



(b)

Fig. 6. Results of anomaly detection for the joint event of the gases, by using KS test setting (a) $\alpha = 0.012$; (b) $\alpha \approx 0.011$. The top curves show the log likelihood and the dots above show the anomalies determined by the algorithm.

jumped by 1.7 ppm, while the average difference is 0.035 ppm. This is roughly a 5000% increase in growth rate of CO, which is clearly anomalous.

For investigating joint events, we use the network structure shown in Fig. 3 with $\mathbf{A}$ as the data from CO_2, $\mathbf{B}$ as data from CO and $\mathbf{C}$ as the data from O_2. Figure 6(a) shows the inference results for the joint event of CO_2-CO-O_2 using this network. We used a history of 5 data points, that is, $m = 5$ in Fig. 3. We have conducted experiments with different m values from $m = 2$ to $m = d = 20$, finding that the inference results do not vary much. The anomalies were again identified by employing the KS test using extreme value distribution with $\alpha = 0.012$.

The results show many interesting features, which we list below in order of dates:

1. 14th July: No joint event was detected on this date, while Fig. 5(b) showed one event identified in the CO data on this date. Thus, while the jump in CO concentration at this time would be enough to trigger an alarm for the single gas system, it was not significant enough within the context of the combined gases. This is an example of an "explained" event.
2. 18th July: Three distinctive joint events were identified around this date. However, Fig. 5 (a)-(c) showed that no anomalies were detected in the CO_2 and O_2 data, and only one event was identified in the CO data. The second and third events would exemplify as "implicit" events.
3. 21st July: Four events closely following one another were identified around this date. In the individual gases, only one event is identified in each gas.
4. 22nd July: The first event is similar to that of the situation on 18th July, where only CO data were anomalous. The second event is where CO concentration dropped, CO_2 increased, both causing an anomaly (while at the same time O_2 dropped slightly but not enough to cause an anomaly). This is an example of "explained" event.
5. 24th July: This is an example of the "group" event where all gases had an anomaly detected and the joint event was observed as well.

We noted previously that there are no anomalies of real concern in this data set. To demonstrate the flexibility of the method, we set α for the KS test at 0.012, which is a large value for anomaly detection in this context. In [17] for example, the authors needed to set $\alpha = 0.00001$ to decrease the false alarm rate. Figure 6(b) shows the anomalies found using $\alpha \approx 0.011$ in order to demonstrate this value can be adjusted by operators at a mine site to identify anomalies specific to a mine.

The results shown above are that of "joint temporal anomaly" detection as discussed in Sec. 3.2, in which where the different gas sensor data are from the same location. It is more important for a sensor network to detect anomalies on a network level, such as the problem described in Sec. 3.3. Unfortunately, we do not have data taken at the neighbouring sensor nodes. However, in practice, the two spatiotemporal problems are almost identical. That is, the data from CO_2 can be from location 1, while data from CO and O_2 are from location 2. Then, the learning, inference and likelihood calculation are exactly the same. Therefore, the method we presented can be easily ported to groups of sensor nodes at different locations.

6 Conclusion and Future Work

In this paper, we used a combination of dimensionality analysis and a Bayesian Network to learn models for gas data from underground coal mine's sensor networks. We identified and defined new types of events for a sensor network. We showed that the anomalies in the data can be identified through inference of the Bayesian Network. Further, we showed that our model is able to identify events in a combination of sensor data that cannot be identified through simple aggregation. For example, it was demonstrated that anomalous events in individual gas data might be explained if considered jointly with the changes in other gases. Vice versa, a network-wide spatiotemporal

anomaly may be detected even if individual sensor readings were within their thresholds. The application of this approach leads to a reduction in the number of false alarms without compromising the safety of monitored mines.

Let us briefly outline possible spatiotemporal extensions of the approach. First of all, a Bayesian Network corresponding to a physical location (for example, shown in Fig. 3 with three subnets A, B, C) can be extended with extra subnets for each new sensor at the same physical location, e.g. D and E. In this case, dependencies between existing and new subnets can be revealed by methods such as transfer entropy.

Transfer entropy [18] identifies a possible relationship between time series, say A and D, denoted $T_{A\to D}$, by estimating the amount of information that a source A_t provides about the next state of a destination D_{t+1} that was not contained in the k past states of the destination $D_{t-k},\ldots,D_t$. In other words, transfer entropy provides a measure of the predictive influence of one element over another — hence, it may help in finding dependencies between sensor data. The active information storage, a measure of the amount of information in the past of a processes that is used in determining its next state [19] may be used in addition to transfer entropy.

Secondly, a Bayesian Network can include subnets corresponding to different physical locations, for example, $A^{(1)}$ and $B^{(1)}$ for location 1, and $B^{(2)}$ and $C^{(2)}$ for location 2, where A, B, C are different gases. In such a case, there may be a temporal dependency between A and B relevant to location 1, a temporal dependency between B and C relevant to location 2, and a spatial dependency between $B^{(1)}$ and $B^{(2)}$. Our approach easily handles situations like this, provided that spatial and temporal dependencies are identified by methods such as transfer entropy. The challenge, however, is in preventing long chains of dependencies spanning the whole sensor network. To address this challenge, an information threshold $\bar{T}$ can be used to distinguish between different transfer entropies. For example, transfer entropy $T_{B^{(1)}\to B^{(2)}} \geq \bar{T}$ would indicate a need to include spatial dependency between $B^{(1)}$ and $B^{(2)}$, while $T_{C^{(2)}\to C^{(3)}} < \bar{T}$ would indicate that there is no need to include a dependency between $C^{(2)}$ and $C^{(3)}$ — thus, breaking a potential chain.

Constructing Bayesian Networks that correspond to dominant information flows in a sensor network is a subject of future research.

Acknowledgement

We would like to thank Greg Rowan and Russell Packham for their generous help with understanding gas behaviour in underground coal mines. We would also like to thank Olivier Fillon and Kerstin Haustein for their help with the data. We would also like to thank the reviewers for their comments that helped to make this a better paper.

References

1. Wang, P., Wang, X.R., Guo, Y., Gerasimov, V., Prokopenko, M., Fillon, O., Haustein, K., Rowan, G.: Anomaly detection in coal-mining sensor data, report 2: Feasibility study and demonstration. Technical Report 07/084, CSIRO, ICT Centre (2007)

2. Xue, W., Luo, Q., Chen, L., Liu, Y.: Contour map matching for event detection in sensor networks. In: Proceedings of the 2006 ACM SIGMOD international conference on Management of data, Chicago, IL, USA, pp. 145–156 (2006)
3. Ypma, A., Duin, R.P.W.: Novelty detection using self-organizing maps. In: Progress in Connectionist-Based Information Systems, vol. 2, pp. 1322–1325. Springer, London (1997)
4. Davy, M., Desobry, F., Gretton, A., Doncarli, C.: An online support vector machine for abnormal events detection. Signal Processing 86(8), 2009–2025 (2006)
5. Mamei, M., Nagpal, R.: Macro programming through Bayesian Networks: Distributed inference and anomaly detection. In: PerCom 2007. Fifth Annual IEEE International Conference on Pervasive Computing and Communications, Los Alamitos, CA, USA, pp. 87–96 (2007)
6. Kumar, A.V.U.P., Reddy, A.M.V, Janakiram, D.: Distributed collaboration for event detection in wireless sensor networks. In: Proceedings of the 3rd international workshop on Middleware for pervasive and ad-hoc computing, pp. 1–8 (2005)
7. Jiao, B., Son, S.H., Stankovic, J.A.: GEM: Generic event service middleware for wireless sensor networks. In: Second International Workshop on Networked Sensing Systems (2005)
8. Friedman, N., Koller, D.: Being Bayesian about network structure: A Bayesian approach to structure discovery in Bayesian Networks. Machine Learning 50, 95–126 (2003)
9. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian network classifiers. Machine Learning 29, 131–163 (1997)
10. Jensen, F.V.: Bayesian Networks and Decision Graphs. Springer, New York (2001)
11. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann, San Francisco (1988)
12. Packard, N.H., Crutchfield, J.P., Farmer, J.D., Shaw, R.S.: Geometry from a time series. Phys. Rev. Lett. 45(9), 712–716 (1980)
13. MacKay, D.J.: Information Theory, Learning and Inference. Cambridge University Press, Cambridge (2003)
14. Myung, I.J.: Tutorial on maximum likelihood estimation. Journal of Mathematical Psychology 47, 90–100 (2003)
15. Fraser, A.M., Swinney, H.L.: Independent coordinates for strange attractors from mutual information. Phys. Rev. A 33(2), 1134–1140 (1986)
16. Massey, F.J.: The Kolmogorov-Smirnov test for goodness of fit. Journal of the American Statistical Association 46(253), 68–78 (1951)
17. Menzies, T., Allen, D., Orrego, A.: Bayesian anomaly detection. In: ICML 2006. Workshop on Machine learning Algorithms for Surveillance and Event Detection, PA, USA (June 2006)
18. Schreiber, T.: Measuring information transfer. Phys. Rev. Lett. 85(2), 461–464 (2000)
19. Lizier, J.T., Prokopenko, M., Zomaya, A.Y.: Detecting non-trivial computation in complex dynamics. In: ECAL 2007. Advances in Artificial Life - 9th European Conference on Artificial Life, Lisbon, Portugal. LNCS (LNAI), vol. 4648, pp. 895–904. Springer, Heidelberg (2007)
20. Grassberger, P., Procaccia, I.: Estimation of the Kolmogorov entropy from a chaotic signal. Phys. Rev. A 28(4), 2591–2593 (1983)
21. Takens, F.: Detecting strange attractors in turbulence. Lecture Notes in Mathematics, vol. 898, pp. 366–381. Springer, Heidelberg (1981)
22. Takens, F.: Invariants related to dimension and entropy. In: Atas do 13 Colóquio Brasiliero do Matemática, Rio de Janeiro (1983)
23. Theiler, J.: Spurious dimension from correlation algorithms applied to limited time-series data. Phys. Rev. A 34(3), 2427–2432 (1986)
24. Dhamala, M., Lai, Y.C., Kostelich, E.J.: Analyses of transient chaotic time series. Phys. Rev. E 64(5), 056207–056216 (2001)

25. Kugiumtzis, D., Lillekjendlie, B., Christophersen, N.: Chaotic time series part I: Estimation of some invariant properties in state space. Modeling, Identification and Control 15, 205–224 (1994)

A Dimensionality Analysis

Grassberger and Procaccia [20] showed that the correlation integral of a time series, $C_d(r)$ can be estimated as:

$$C_d(N,r) = \frac{1}{(N-1)N} \sum_{j=1}^{N} \sum_{\substack{i=1 \\ i \neq j}}^{N} \Phi(r - \|\mathbf{y}_i - \mathbf{y}_j\|). \tag{7}$$

Here Φ is the Heaviside function (equal to 0 for negative arguments and 1 otherwise). The vectors $\mathbf{y}_i$ and $\mathbf{y}_j$ contain elements of the observed time series $\{x_t\}$ with the dynamical information in one-dimensional data converted or reconstructed to spatial information in the d-dimensional embedding space $\mathbf{y}$ [21] as presented in Equation 2. The norm $\|\mathbf{y}_i - \mathbf{y}_j\|$ is the distance between the vectors in the d-dimensional space, e.g., the maximum norm [22]:

$$\|\mathbf{y}_i - \mathbf{y}_j\| = \max_{\tau=0}^{d-1}(x_{i+\tau} - x_{j+\tau}) \tag{8}$$

Put simply, $C_d(r)$ computes the fraction of pairs of vectors in the d-dimensional embedding space that are separated by a distance less than or equal to r. In order to eliminate auto-correlation effects, the vectors in Equation 7 should be chosen to satisfy $|i-j| > L$, for some positive L, and at the very least $i \neq j$ [23].

The correlation dimension ν is found by:

$$\nu = \lim_{r \to 0} \lim_{N \to 0} \frac{\ln C_d(N,r)}{\ln r} \tag{9}$$

That is, within certain ranges of r and d, the correlation integral $C_d(r)$ may be proportional to some power of r, $C_d(r) \sim r^{\nu}$ [20]. If the dynamical process is unfolded by choosing a sufficiently large $d > d_\nu$, a typical slope of the plot $\ln C_d(r)$ versus $\ln r$ becomes independent of d. Thus the common numerical practice of finding the embedding dimension d of the data set is to compute the slope from a linear region of the $C_d(N,r)$ plot. For $d \leq \lfloor \nu \rfloor$, where $\lfloor \nu \rfloor$ denotes the largest integer less than or equal to ν, the slope is equal to d. For $d > \lfloor \nu \rfloor$, the slop saturates at a constant value which is usually taken to be the estimated value of ν [24].

The data is from telemetric sensors, the time difference between two data point is 30 seconds, thus between 3rd July 0100 and 25th July 0556, we have 68194 data samples. To find the embedding dimensions, we divide the data into 1 day periods, this gives us 2880 data points per period. Kugiumtzis *et al.*[25] showed that this is a reasonable number for calculating embedding dimensions. We found the embedding dimension to be $d = 20$.