

Data driven Bayesian network to predict critical alarm

Joseph Mietkiewicz

Hugin Expert A/S / Food Science Environmental Health TU Dublin. E-mail: D21127042@mytudublin.ie

Anders Madsen

Hugin Expert A/S, Department of Computer science Aalborg University, Denmark.

E-mail: anders@hugin.com

Modern industrial plants rely on alarm systems to ensure their safe and effective functioning. Alarms give the operator knowledge about the current state of the industrial plants. Trip alarms indicating a trip event indicate the shutdown of systems. Trip events in power plants can be costly and critical for the running of the operation. This paper demonstrates how trips events based on an alarm log from an offshore gas production can be reliably predicted using a Bayesian network. If a trip event is reliably predicted and the main cause of it is identified, it will allow the operator to prevent it. The Bayesian network model developed to predict trip events is purely data-driven and relies only on historic data from the alarms log from offshore gas production. We describe the method used to build the Bayesian network and the approach used to identify the most key alarm related to the Trip. We then assess the performance of the Bayesian network on the alarm log of an offshore gas production. The preliminary performance results show significant potential in predicting trips and identifying key alarms. The model is developed to support decision-making of a human operator and increase the performance of the plant.

Keywords: Alarm management, Bayesian Network, Prediction of trip event, Time-stamped sequences, causality analysis.

1. Introduction

With the increasing complexity of the industry, the number of alarms increases drastically which can lead to an overload of alarms for a human operator. Wang et al. (2015) gives a review of the status of research and open problem regarding industrial alarm systems. In this paper, we study a way to predict critical events and help the operator identify critical alarms to increase the performance of offshore gas production. One of the problems with alarms data is the high frequency of alarms coming from different sources. Lots of work have been done to study the correlation between alarms to group them. Salah et al. (2013) provide a literature review of the correlation technique. Kargaran et al. (2021) provide a useful and intuitive review of the technique used. Weiß et al. (2018) develop a technique base on hierarchical clustering to identify causal alarm clusters to reduce the number of alarms and avoid overflow. Soares et al. (2016) investigate among other things the use of correlation analysis, PCA, and hierarchical

clustering in a natural gas processing plant to comply with EEMUA recommendation. An other ideas to predict alarms is to use pattern matching. Zhu et al. (2016) use pattern recognition to predict critical alarms. Bayesian network allow to model the process and can help operator making decision. da Silva et al. (2018) explain how Bayesian Network can be implemented in an industrial environment to help an human operator. It is possible to use it for online prediction due to its fast computation time and it can represent causal interaction between alarms. Liang et al. (2019) explain how Bayesian networks can be used with alarms datasets. Kirchhübel and Jørgensen (2019) study the creation of a Bayesian network based on a qualitative causal model. Expert knowledge are really valuable to build Bayesian network but it is not always accessible and it can be a long process to collect it. The idea to this paper is to build a data-Driven Bayesian network to predict critical alarm an provide root cause analysis to an operator. The paper is structure as follow: We introduce background about Bayesian Network,

Proceedings of the 32nd European Safety and Reliability Conference.

Edited by Maria Chiara Leva, Edoardo Patelli, Luca Podofillini and Simon Wilson

Copyright © 2022 by ESREL2022 Organizers. *Published by* Research Publishing, Singapore

ISBN: 981-973-0000-00-0 :: doi: 10.3850/981-973-0000-00-0_output

2 First Author and Second Author

how to use them to predict alarm and identify probable root cause. We then apply the described method to a case study.

2. Background

The data set from an alarm log from an off shore gas production is study. The wellhead is the component that links the extracted gas to the production system. If the pipeline from the wellhead is closed the flow of gas is stopped. We analyze the alarm that indicate a Trip of the shutdown valves. These valves stop the incoming flow of gas. It takes on average 23 minutes to recover from this trip. If it can be predicted in advance and given to the operator all the other alarms linked to it, this shutdown can be avoided. We develop a method to build a Bayesian Network that take into account each alarm and compute the probability of critical alarms at any time and provide root cause analysis. In this section we introduce the background needed to build and use the Bayesian Network. We first introduce Bayesian Networks and then how to use them for prediction and root cause analysis.

2.1. Bayesian Networks

A Bayesian network is a direct acylique graph where the node represents variables and the arc the causal interaction between those variables. It is a powerful tool to model causal interaction and predict events. A Bayesian Network is define as follows:

Definition Jensen and Nielsen (2007): A discrete Bayesian Network $\mathcal{N} = (\mathcal{A}, \mathcal{G}, \mathcal{P})$ consist of

- A DAG $\mathcal{G} = (V, E)$ with nodes $N = N_1, \dots, N_n$ and directed links $E = (v_1, \dots, v_n)$
- A set of discrete random variables, $\mathcal{A} = (A_1, \dots, A_n)$, represented by the nodes of \mathcal{G}
- A set of conditional probability distribution, \mathcal{P} , containing one distribution $P(A_N | A_{pa(N)})$, for each random variable $A_N \in \mathcal{A}$

A Bayesian Network encodes a joint probability distribution over a set of random variables, \mathcal{A} , of

a problem domain. The set of a conditional probability distribution, \mathcal{P} , specifies a multiplicative factorization of the joint probability distribution over \mathcal{A} .

$$P(\mathcal{A}) = \prod_{(A \in N)} P(A_N | A_{pa(N)}) \quad (1)$$

2.2. Naive Bayes

In the study we the Naive Bayes classifier. The Naive Bayes is a simple and power full model to predict an event. One of the advantages of Naive Bayes is its robustness to noise. It has the strong assumption of independence between all the variables given the class variable.

Lets A_1 the alarm to predict and A_2, \dots, A_n the other alarms. The joint probability distribution over the alarms can be written as:

$$P(\mathcal{A}) = P(A_1) \prod_{(A \in N)} P(A_N | A_1) \quad (2)$$

Lets (a_2, \dots, a_n) be an instance to classified and a_1 the class of the instance. The joint probability distribution for Naive Bayes can be written as:

$$P(a_2, \dots, a_n | a_1) = \prod_{i=2}^n P(a_i | a_1) \quad (3)$$

The model is train and the joint probability estimate on the data. When the model is use online, each time an alarm trigger the probability of the alarm A_1 is updated using Bayes formula.

Example: Lets A_2 be an alarm:

$$P(A_1 | A_2) = \frac{P(A_2 | A_1)P(A_1)}{P(A_2)} \quad (4)$$

2.3. Tree augmented Naive Bayes

Tree augmented Naive bay add dependencies between variables. A Chow-Lui tree structure is add to the naive Bayes on the feature variables. Lets $A = (A_1, \dots, A_n)$ be the variables and A_1 be the root variable. Tree augmented naive Bayes are build according to the following step

- Calculate the conditional mutual information $I(A_i, A_j)$
- Build a maximal-weight spanning tree T_S for the complete I-weighted graph over $A \setminus A_1$ and direct edges away from it.
- Direct edges in T_S by selecting a root in $A \setminus A_1$ and direct edges away from it.
- Construct the graph corresponding to a Naive Bayes model over A with A_1 as root and add edges of T_S to it

2.4. Sensitivity Analysis

Sensitivity analysis as described in Kjaerulff and Madsen (2008) can be used to prioritize alarms link to the trip.

Sensitivity analysis is used to identified critical alarms link to the trip. The key alarm can be found by using the Normalized likelihood (NL).

$$\begin{aligned} NL &= \frac{P(\epsilon|x)}{P(\epsilon)} = \frac{P(\epsilon', x)/P(x)}{P(\epsilon')} \\ &= \frac{P(x|\epsilon')P(\epsilon')/P(x)}{P(\epsilon')} = \frac{P(x|\epsilon')}{x} \end{aligned} \quad (5)$$

Each normalized likelihood is a measure of the impact of a subset of evidence on the hypothesis. By comparing the normalized likelihoods of different subsets of the evidence, we compare the impact of the subset of evidence on the hypothesis.

The Cost of omission can also be used to determine if an alarm is important or not regarding an other alarm A_1 . The cost of omission measure the impact between if the alarms is present or not. A low cost of omission mean that the alarms don't have an important impact on A_1 . This measurement can be used to select the most important alarms in case of an emergency situation.

$$c(P(X|\epsilon), P(X|\epsilon \setminus \epsilon_i)) = \sum_{x \in \text{dom}(x)} P(x|\epsilon) \log \frac{P(x|\epsilon)}{P(x|\epsilon \setminus \epsilon_i)} \quad (6)$$

The impact of an alarm on A_1 can also be study and determine if this alarm have a valuable infor-

mation regarding A_1 . We first define the distance $d(p, q)$ between two probabilities. For $p \neq q$:

$$d(p, q) = \left| \frac{q}{p} - 1 \right| \quad (7)$$

This distance is useful to compare the impact of an alarm on the probability of the alarm to predict. This distance is use to define if an alarm is important, redundant given other alarms or irrelevant regarding A_1 .

Lets $a = (a_2, \dots, a_n)$ be the alarms at a time T , A_1 the alarm to predict and δ a positive threshold value.

An alarm a_i is important if the difference between the probability $q = P(A_1|a \setminus a_i)$ and $p = P(A_1|a)$ are not almost equal: $d(p, q) > \delta$

An alarm a_i can be categorize as redundant if $p = P(A_1|a \setminus a_i)$ is almost equal to $q = P(A_1|a)$. That is $d(p, q) < \delta$

An alarm is irrelevant if $q = P(A_1|a' \setminus a_i)$ is almost equal to $p = P(A_1|a)$ for all subset $a' \in a$. That is $d(p, q) < \delta$

A subset of alarm $a' \subset a$ is sufficient if $q = P(A_1|a')$ is almost equal to $p = P(A_1|a)$. That is $d(p, q) < \delta$.

With those measurement the most important set of alarms related to the one to predict can be found and The importance of the alarm in the set can be measure.

3. Methodology

The data set are transom to fit Bayesian network. From the list of event in the alarm log we create a data set composed of dummy variable. Each time an alarm tiger the alarms have value 1 if they trigger 0 else. For each alarm A_i we associate a binary vector $s_i(t)$.

4 First Author and Second Author

Dealing with alarm data set two representation of it can be used one using the return to normal one only using the trigger time.

The first representation use only the time when the alarm trigger. Lets $T = ((t_1, r_1), \dots, (t_n, r_n))$ the vector of the trigger tie and return to normal of the alarm A_1 .

$$s_i(t) = \begin{cases} 1 & \exists i \in \mathbb{N}, i < n, t = t_i \\ 0 & \text{else} \end{cases} \quad (8)$$

The second representation use the return to normal. The vector have value 1 from the trigger of the alarm until it return to normal.

$$s_i(t) = \begin{cases} 1 & \exists i \in \mathbb{N}, i < n, t_i < t < r_i \\ 0 & \text{else} \end{cases} \quad (9)$$

As mention in Yang et al. (2013) the second formulation can lead to overestimate correlation between alarms. Similar issue happened with Bayesian network. This formulation can lead to misleading CPT and over estimate the probability of the class variable A_1 .

To be able to predict the alarm A_1 , we add 1 before the event happened in the the training data. Lets $T = (t_1, \dots, t_c)$ be the vector of time when the alarm A_1 trigger and $s_1(t)$ the vector of the value of A_1 in the dataset.

$$s_1(t) = \begin{cases} 1 & \text{if } \exists t_i \in T, t_i - T_c < t < t_i \text{ and } s_1(t_i) = 1 \\ 0 & \text{else} \end{cases} \quad (10)$$

To add dependencies between all the alarms we extend the period of the alarm of the time T_c after they happened. For $i \neq 1$, lets $T = (t_1, \dots, t_c)$ be the vector of time when the alarm A_1 trigger and $s_1(t)$ the vector of the value of A_1 in the data set.

$$s_i(t) = \begin{cases} 1 & \text{if } \exists t_i \in T, t_i < t < t_i + T_c \text{ and } s_i(t_i) = 1 \\ 0 & \text{else} \end{cases} \quad (11)$$

In this way, during the training phase, the model can learn the dependence between the alarm hap-

pening before and close to A_1 since they have both value 1.

Table 1. Example of the TimeStamp and Source of the dataset

TimeStamps	Source
2022-01-18 16:24:23	<i>A2.HH</i>
2022-01-18 16:29:50	<i>A3.HI</i>
2022-01-18 18:23:43	<i>A1.TRIP</i>

The data is then preprocess. The first step is to remove chattering alarms. We simply remove the 12 first alarms base on their frequency. The new dataset is composed of 253958 trips of alarms after removal of return to normal and acknowledgment events. Each alarm can have multiple condition name: Trip, HH,HI,LO,LL,ANS+,ANS-,IOP,IOP-,... that indicate different information. To not mix those condition we add to the name of each alarms his specific condition name when those alarm append. For example if the name of the alarm is A_8 with condition name LO, the variable $A_8.LO$ is created. We convert the data as discuss previously. This conversion is illustrated by the table1 to table 2.

4. Evaluation on a Case study

The data is composed of the record of the alarm log for 1 month from 18 January 2022 to 21 February 2022. 2182676 events were made from 2621 different alarms. Those events can be decomposed into 3 different kinds, the trip of the alarm, acknowledgment of the alarms, and return to normal. From the 2182676 events, there are 1083324 trigger of alarms.

4.1. Prediction

Since only 14 shutdowns of the valves have been identified we use the leave one out strategy to assess the performance of the Bayesian networks. We create the train set by removing one trip of the shutdown valves and all the alarms that append before until the previous trip. We then build the

Table 2. Process version of the example data (table 1)

TimeStamps	$A_2.HH$	$A_3.HI$	$A_1.TRIP$
2022-01-18 16:24:23	0	1	0
2022-01-18 16:29:50	1	0	0
2022-01-18 18:23:43	0	0	1

Table 3. Example on the alarms in table 1 of the Dataset use for train TAN. The alarm 003XZV003 is the one to predict

TimeStamps	$A_2.HH$	$A_3.HI$	$A_1.TRIP$
2022-01-18 15:26:33	0	1	0
2022-01-18 16:24:23	1	1	1
2022-01-18 16:29:50	1	1	1
2022-01-18 18:23:43	1	0	1

model and train it with the training data. We use HUGIN software to build and test the Bayesian networks.

Algorithm 1 Prediction Trip event

for Each Trip event instance **do**
 $data_{test}$ =data of one Trip event
 $data_{train}$ = all the others
 step 2: Train and build the NB
 step 4: Predict Trip event using BN
 step 5: Identified key alarms
end for

The naive Bayes model predict all the trip but add some false prediction. The use of TAN remove those false prediction. This is due to strong correlation between some alarms (for example 026BPZI065.HI and 026BPZI065.HH, with Naive Bayes they are considered as two independent alarms). The models predicts at 99% the chance of the Trip of the shutdown valves on average 15 seconds before it happened.

4.2. Sensitivity analysis

The sensitivity analysis can be done at any time. We choose to do it at the first the probability of the

alarm A_1 is superior at 50% and when this alarm trigger.

13 second before the alarm A_1 trigger the probability of A_1 increase at 89%. At this moment we operate the sensitivity analysis. The constant for importance is set at 0.1. With this value will select the sub set of alarm base on the cost of omission that is sufficient for the probability of the Trip.

We try to maximise the cost of omission to select as less variable to be possible but keeping the sub set of alarm sufficient.

At this first time 58 alarm are sufficient among them only 2 are important.

Table 4. Cost of omission

Source distance	NL	cost of omission
$A_2.ALM$	4.70	709.7
$A_6.ALM$	5.35	330.3

At time of the Trip there is only 5 variable that are sufficient. The alarm are in this case are redundant.

6 *First Author and Second Author*

Table 5. Cost of omission

Source distance	NL	cost of omission
A_2 .ALM	4.70	0.002
A_3 .LO	3.7	0.02
A_4 .ALM	1.4	0.003
A_5 .HI	3.8	0.0004
A_6 .ALM	8.2	0.0002

The alarm A_2 is an alarm in the compressor system. The alarm A_3, A_4, A_5 are from the production separation. and A_6 an alarm from the wellhead. We can deduce that the upset start from the compressor system propagate to the production separation and finally cause a Trip of the shutdown valve of the wellhead.

5. Conclusion

Using a Bayesian network to predict the shutdown of the production of the power plant gives a promising result. This Bayesian Network can be used for online prediction and help the operator make decisions. With Bayesian Network the cause of this Trip can be study. In future works, we will study algorithms to learn Bayesian Networks more specific to alarm data. We will focusing particularly in finding causality between alarms. The construction of the Bayesian network will benefit greatly from the knowledge of the experts.

Acknowledgement

We would like to thank Rob Turner and Afshar Puya from Yokogawa for providing the dataset and giving us a better insight into it. This work has been done within the collaborative intelligence for safety Critical systems project (CISC). The CISC project has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie grant agreement no. 955901.

References

- da Silva, M. J., C. E. Pereira, and M. Götz (2018). Context-aware recommendation for industrial alarm system. *IFAC-PapersOnLine* 51(11), 229–234.
- Jensen, F. V. and T. D. Nielsen (2007). *Bayesian networks and decision graphs*, Volume 2. Springer.

- Kargaran, A. H., A. Neshastegaran, I. Izadi, and E. Yazdian (2021). Analytical derivation and comparison of alarm similarity measures. *IFAC-PapersOnLine* 54(3), 360–365.
- Kirchhübel, D. and T. M. Jørgensen (2019). Generating diagnostic bayesian networks from qualitative causal models. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1239–1242. IEEE.
- Kjaerulff, U. B. and A. L. Madsen (2008). Bayesian networks and influence diagrams. *Springer Science+ Business Media* 200, 114.
- Liang, R., F. Liu, J. Qu, and Z. Zhang (2019). A bayesian-based self-diagnosis approach for alarm prognosis in communication networks. In *2019 8th International Symposium on Next Generation Electronics (ISNE)*, pp. 1–3. IEEE.
- Salah, S., G. Maciá-Fernández, and J. E. Díaz-Verdejo (2013). A model-based survey of alert correlation techniques. *Computer Networks* 57(5), 1289–1317.
- Soares, V. B., J. C. Pinto, and M. B. de Souza Jr (2016). Alarm management practices in natural gas processing plants. *Control Engineering Practice* 55, 185–196.
- Wang, J., F. Yang, T. Chen, and S. L. Shah (2015). An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems. *IEEE Transactions on Automation Science and Engineering* 13(2), 1045–1061.
- Weiß, I., J. Kinghorst, T. Kröger, M. F. Pirehgalin, and B. Vogel-Heuser (2018). Alarm flood analysis by hierarchical clustering of the probabilistic dependency between alarms. In *2018 IEEE 16th International Conference on Industrial Informatics (INDIN)*, pp. 227–232. IEEE.
- Yang, Z., J. Wang, and T. Chen (2013). Detection of correlated alarms based on similarity coefficients of binary data. *IEEE Transactions on Automation science and Engineering* 10(4), 1014–1025.
- Zhu, J., C. Wang, C. Li, X. Gao, and J. Zhao (2016). Dynamic alarm prediction for critical alarms using a probabilistic model. *Chinese Journal of Chemical Engineering* 24(7), 881–885.

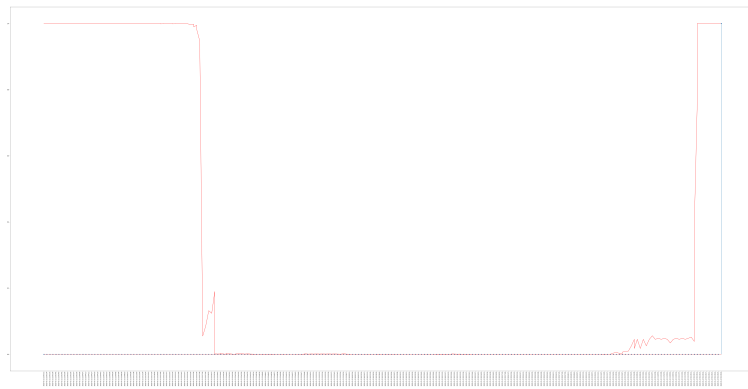


Fig. 1. Prediction of TRIP of the wellhead with TAN. In red the probability of Trip and blue the real TRIP