

A Bayesian Approach for Maintenance Action Recommendation

Vassilis Katsouros¹, Vassilis Papavassiliou¹, and Christos Emmanouilidis²

¹*Institute for Language and Speech Processing, Athena Research and Innovation Center, Athens, GR 15125, Greece*

*vsk@ilsp.gr
vpapa@ilsp.gr*

²*Institute for Language and Speech Processing, Athena Research and Innovation Center, Xanthi, GR 67100, Greece*

chrise@ceti.gr

ABSTRACT

This paper presents a Bayesian approach for maintenance action recommendation tested on the PHM 2013 Data Challenge dataset. The Challenge focused on maintenance action recommendation based on historical cases and the algorithms were evaluated on their ability to recommend confirmed problem types. The proposed approach is based on a Bayesian inference methodology and deals with recommending an already known problem type for each case. The recommender can be viewed as a classifier among the confirmed problem types. For each such problem type class the a priori probabilities for the events which characterize the problem type from the training data are estimated. When testing cases are presented, the recommender calculates the a posteriori probabilities for each of the confirmed problem types and suggests the type of problem that corresponds to the maximum a posteriori (MAP) probability.

1. INTRODUCTION

In the recent years, Prognostics and Health Management (PHM) is an area that has attracted the attention of both academia and industry. It is gradually recognized as a key pillar in implementing Condition – Based Maintenance Strategies (CBM), which are aimed at improving the efficiency of Asset Lifecycle Management practices.

Significant research results have been emerged with the development of sophisticated algorithms and advanced solutions for system detection, diagnosis and prognosis. PHM paradigms typically involve a training stage, where algorithms are used to discover and learn patterns and trends from historical data followed by a testing stage, where trained models are applied to new machine condition data for health assessment and/or prediction.

Whereas a multitude of such methodologies have emerged and were applied in various problems and domains, a lack of common ground regarding performance evaluation is hampering efforts to establish a better understanding of the efficiency and potential impact of PHM methodologies. This lack was related to both the availability of common benchmarking datasets, as well as to the adoption of specific evaluation metrics. The former is being addressed in the last few years with the production and distribution of publicly available benchmarking datasets, often the result of focused research efforts to establish prototype test beds for obtaining such data (Nectoux et al., 2012). The latter has given rise to a debate as per what would constitute adequate performance evaluation metrics for different aspects of the PHM problem domain (Eker et al., 2012), (Zhou et al., 2013).

There is a diverse range of PHM methodologies based on machine learning techniques that have been employed for health assessment and prediction from multiple sensorial data. Among these techniques Bayesian approaches provide an elegant and theoretically sound framework for estimating the expectancy of an event to belong to the learned class by modeling the interdependencies among the attributes of the historical sensor data.

In the last decade Bayesian approaches have been applied in tackling various tasks of CBM and PHM. Elnahrawy and Nath (2004) employed Bayesian classifiers to learn contextual information and then make inference in sensor networks. Saha and Goebel (2008) also use Bayesian techniques for statistical modeling of operational conditions to provide estimates of remaining useful life (RUL) in the form of a probability density function. More recently, Karandikar, Abbas and Schmitz (2013) applied Bayesian inference to estimate the RUL for selected tools. The implementation of Naïve Bayes classifiers for early detection of developing catastrophic failures in machining operations and its integration a machine tool control system was also recently proposed (Mehta et al., 2013).

Vassilis Katsouros et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

In the present work, we adopt Bayesian inference to implement a recommender that was submitted to the 2013 PHM Data Challenge, employing MAP probability classification of test cases into the known problem types.

The rest of the paper is organized as follows. Section 2 described the problem formulation, the training and test datasets and introduces the employed notation. Section 3 explains the approach of the Bayesian recommender. In Section 4 we discuss the experimental results and the evaluation of the suggested algorithm and, finally, we present the conclusions in Section 5.

2. PROBLEM FORMULATION AND DATASETS

In this section we present the problem formulation for the task of the PHM Data Challenge 2013, the available datasets and the performance evaluation method.

The task of the Data Challenge 2013 was to implement a recommender which takes as input records of a case, decides whether this case is a nuisance or problem, and in the latter case recommends also the problem type from a known set of problems emerged from historical data.

A case consists of a collection of event codes each of which corresponds to a number of parameters. This data was generated automatically the equipment monitoring system. Every time a specific condition is met onboard, the control system generates a specific event code and takes a snapshot of all of the parameters that are measured onboard. On the other hand, cases have been created either manually by an engineer or automatically by a control system. A record of a case therefore can be defined as a single event code along with the respective measurements of the parameters. There were 30 parameters of onboard measurements, recorded each time an event code was generated.

The Data Challenge provided a dataset of cases with their event codes and respective parameters for training and another one for testing/evaluation. The training set included the classification of the cases into nuisance or problem. For the cases classified as problems, the corresponding problem label/identifier was also provided. It must be noted that due to proprietary concerns a detailed description of the data and the domain was not provided. The training data involved in total 1.316.653 records that correspond to 10.459 cases of which 10.295 were characterized as nuisance and 164 as problem, resulting into 13 distinct problem identifiers/codes. The testing dataset involved in total 1.893.882 records of event codes – measured parameters that correspond to 9.358 distinct cases. The ground truth of the testing dataset involved 174 problem cases, with the remaining 9.184 being nuisance cases. A recommender should identify the problem cases from the 9.358 testing cases and for each one of them provide the respective problem identifier.

The evaluation of the proposed algorithms was carried out on a set of cases that involved all 174 ground truth problem

cases and a random selection of 174 from the total of 9.184 nuisance cases. The evaluation of the proposed algorithms was based on a score that was calculated by subtracting from the number of outputs, the number of wrongly identified problems as well as the number of nuisance cases. In particular, the ranking of the submitted recommenders was carried out using the following formula:

$$\text{Score} = \# \text{Outputs} - \# \text{Incorrect Outputs} - \# \text{Nuisance Outputs} \quad (1)$$

Note that the maximum number of outputs that a recommender could achieve is 348, i.e. the sum of 174 ground truth problem cases and 174 nuisance cases. Therefore, a recommender should identify on one hand the correct problem types of the test cases and on the other hand discard nuisance cases.

Let us now introduce some notation that will assist us in the mathematical definition of the problem. We denote by $\mathbf{C} = \{C_1, C_2, \dots, C_N\}$ the set of cases, where the index indicates the unique identifier of each case. For each case C_i , the corresponding events are denoted by $E_j^{C_i}$, where j ranges from 1 up to the number of events N^{C_i} that comprise the case C_i and the respective parameter measurements of an event $E_j^{C_i}$ are denoted by $\mathbf{m}_j^{C_i} \in \square^n$ where n is the dimension of the parameter space. As mentioned above, the event codes can take values from a finite set that eventually captures all possible conditions met onboard, denoted by $\mathbf{E} = \{E_1, E_2, \dots, E_N\}$, with N being the cardinality of the event codes set. It must also be noted that some of the parameter measurements might be missing due to possible failure of the equipment or other unknown factors. We define with $\mathbf{P}^L = \{P_0^L, P_1^L, P_2^L, \dots, P_M^L\}$ the set of all possible problem types that one may encounter, with the problem type P_0^L being the label/identifier used for the nuisance cases. We define the training dataset of cases-event codes-parameters by \mathbf{D}^{tr} that consists of tuples of the form $(C_i^{tr}, E_j^{C_i^{tr}}, \mathbf{m}_j^{C_i^{tr}})$, where the tr subscript in C_i^{tr} denotes that the case belongs to the training dataset. The training dataset of cases-problem types is denoted by \mathbf{U}^{tr} and consists of pairs of the form $(C_i^{tr}, P^{C_i^{tr}})$, where $P^{C_i^{tr}} \in \mathbf{P}^L$. In a similar manner we define the testing dataset by \mathbf{D}^{te} . The aim of a recommender is to produce the set \mathbf{U}^{te} which includes the pairs of testing cases-problem types of the form $(C_i^{te}, P^{C_i^{te}})$, where C_i^{te} is the i -th test case and $P^{C_i^{te}}$ the recommended problem type. A

test case C_i^{te} may be assigned a problem type label $P_i^{C_i^{tr}} \in \mathbf{P}^L$. There are two types of errors of a recommender: (a) Type I errors have to do with cases that although correctly identified as corresponding to a problem, they have been assigned a wrong problem type. (b) Type II errors have to do with cases which do not correspond to any problem, but they have been assigned a problem type, i.e. nuisance cases.

3. THE BAYESIAN RECOMMENDER

Our approach is based on Bayesian classification, in which we define distinct problem classes one for each problem type, calculate the posterior probability of a test case for each problem class and subsequently recommend the problem class that corresponds to the maximum a posteriori (MAP) probability.

Bayesian inference provides a rigorous mathematical framework for updating our belief for the occurrence of an unknown random variable when new information on dependent random events or variables becomes available. In particular, the Bayes rule relates the prior probability of a random event with the likelihood obtained from experimental evidence in order to determine the posterior probability about the random event after observing the experimental results. We refer the reader to Koller & Friedman (2009) for an extensive study of the theoretical foundations of Bayesian techniques and inference on Probabilistic Graphical Models.

It must be noted that we treat only the problem of assigning a test case to a problem type without caring for nuisance cases or, to put it another way, our recommender would never assigns the nuisance label P_0^L to any of the test cases.

Each case can be represented by the set of distinct event codes that are included in it and is also assigned to a problem type/class. We therefore calculate from the training dataset the number of occurrences that each event code is encountered given the respective problem type. Let us define the matrix \mathbf{N}_{EP}^{tr} of dimension $N \times M$

$$\mathbf{N}_{EP}^{tr} = [n_{ij}], \quad i = 1, \dots, N, \quad j = 1, \dots, M \quad (2)$$

where N is the number of distinct event codes of the set \mathbf{E} and M the number of problem types of the set $\mathbf{P}^L \setminus \{P_0^L\}$, i.e., we ignore the nuisance class. In practice, we use as N not the total number of event codes that one may observe in the training set but we restrict N to be the number of event codes of only the training cases that have been classified as problems.

The elements n_{ij} of the matrix \mathbf{N}_{EP}^{tr} represent the counting measure of how many times in the training dataset one encounters a record of a case that is classified to problem type j and at the same time involves the event code i . Summing over the columns of the matrix \mathbf{N}_{EP}^{tr} as follows,

$$N_j = \sum_{i=1}^N n_{ij}, \quad j = 1, \dots, M, \quad (3)$$

one obtains the number of records from the training set that have been assigned to each problem type. Dividing each element n_{ij} of the matrix \mathbf{N}_{EP}^{tr} by the respective number of records N_j from the training set that have been assigned to problem types that corresponds to the column j

$$p_{ij} = \frac{n_{ij}}{N_j}, \quad i = 1, \dots, N, \quad j = 1, \dots, M \quad (4)$$

one obtains an estimate for the conditional probabilities of observing a case that includes an event code E_i given that this case has been classified in the problem type P_j , i.e.

$$\Pr[E_i | P_j] = p_{ij}, \quad i = 1, \dots, N, \quad j = 1, \dots, M \quad (5)$$

In order to avoid zero values for the p_{ij} , we replace zeros with a small positive number ε , with a typical value of 10^{-10} . In addition, one may estimate from the training data the prior probabilities of each problem type by using the following expression

$$\Pr[P_j] = \frac{N_j}{\sum_{i=1}^M N_i}, \quad j = 1, \dots, M \quad (6)$$

Now, given a test case C_k we would like to calculate for each problem type P_j the respective conditional probability

$$\Pr[P_j | C_k], \quad j = 1, \dots, M, \quad (7)$$

and decide about the problem type P_k of the case using the maximum a posteriori (MAP) probability, as follows

$$P_k = \arg \max_{j=1, \dots, M} \Pr[P_j | C_k], \quad (8)$$

Applying the Bayes rule to Eq. (7) and replacing into Eq. (8) we obtain

$$P_k = \arg \max_{j=1, \dots, M} \{ \Pr[C_k | P_j] \Pr[P_j] \} \quad (9)$$

disregarding the constant term $\Pr[C_k]$.

Since each case consists of a number of event codes, assuming independence among the event codes, we may approximate the first term in the right hand side of Eq. (9) by

$$\Pr[C_k | P_j] = \prod_{i=1}^{N_{C_k}} \Pr[E_i^{C_k} | P_j] \quad (10)$$

where N_{C_k} is the number of event codes of the test case C_k .

In order to avoid numerical overflow we replace the product of probabilities of Eqs. (9) and (10) with the sum of the respective logarithms as follows

$$P_k = \arg \max_{j=1, \dots, M} \{ \log \Pr[C_k | P_j] + \log \Pr[P_j] \} \quad (11)$$

and

$$\log \Pr[C_k | P_j] = \sum_{i=1}^{N_{C_k}} \log \Pr[E_i^{C_k} | P_j] \quad (12)$$

Therefore, by using Eq. (5) and (6) in (11) and (12) we find the problem type that gives the MAP probability for the test case C_k under consideration.

4. EXPERIMENTAL RESULTS AND EVALUATION

The approach presented in this paper focuses of the task of identifying the correct problem type of a case without taking care whether this case is nuisance. In this section we present the experimental results and discuss the performance of the suggested approach.

The training dataset includes in total 164 cases classified into 13 problem types. The ground truth of the test dataset, provided after the closure of the competition, includes in total 174 cases categorized in 14 problem types. The distribution of cases in problem type categories for the training and test datasets is shown in Table 1. One can observe that the training dataset includes one case with a problem type identifier (P7940) that is not found in the test dataset and the test dataset includes cases that have been categorized in two problem types (P0932 and P6880) for which no training data is available.

	Training Dataset	Test Dataset
Problem Type ID	Number of Cases	Number of Cases
P0159	19	15
P0898	4	6
P0932	-	2
P1737	2	2
P2584	53	26
P2651	13	13
P3600	17	20
P6559	3	1
P6880	-	15
P7547	6	4
P7695	17	37
P7940	1	-
P9766	14	12
P9965	2	5
P9975	13	16
Total	164	174

Table 1. Distribution of cases in problem type categories for the training and test datasets

Initially we tested the approach on the training dataset, since the ground truth for the test dataset was not available. We relaxed the requirement for the algorithm to suggest only one problem type and examined the sorted list of problem types in descending order of their MAP probabilities. In the remainder, we refer to the top-k choice, when the correct problem type is found in the sorted list of the recommender within the first k problem types. The overall performance of the recommender for the top-1 choice of the problem type was 86.59%, for the top-3 choice 95.12% and for the top-5 choice 96.95%. The detailed results of the recommender for the training dataset are shown in Table 2. The suggested algorithm was evaluated on the test set of the PHM Data Challenge 2013 and received an overall score of 60 (<http://www.phmsociety.org/events/conference/phm/13/challenge>).

Note that our algorithm does not take into consideration the task of minimizing the number of nuisance cases (Type II errors), and as a result in our score the term #Nuisance outputs in Eq. (1) is equal to 174, i.e. the total number of nuisance terms of the test set. The detailed results of the suggested algorithm are presented in Table 3. Note that the percentages of the aggregate score reported in the bottom line of Table 3 have been calculated by dividing the aggregate number of correctly classified cases by 157 and not by 174, i.e. we have excluded the cases of the problem types P0932 and P6880 that were not present in the training dataset and therefore it was not possible to be predicted by any recommender.

Number of Correctly Classified Cases			
Problem Type ID	Top 1	Top 3	Top 5
P0159	15	16	18
P0898	4	4	4
P1737	1	1	1
P2584	49	53	53
P2651	13	13	13
P3600	15	17	17
P6559	1	2	2
P7547	2	5	5
P7695	15	17	17
P7940	1	1	1
P9766	14	14	14
P9965	2	2	2
P9975	10	11	12
Total	142	156	159

Table 2. Number of training cases correctly classified for each problem type.

Problem ID	Type	Top 1	Top 2	Top 3	Top 4	Top 5
P0159		1	2	4	7	10
P0898		0	1	1	5	6
P1737		0	0	0	0	0
P2584		12	19	21	24	25
P2651		6	7	7	7	11
P3600		9	15	18	19	19
P6559		0	0	0	0	0
P7547		1	1	1	1	1
P7695		24	30	32	35	37
P9766		5	5	5	8	9
P9965		0	0	0	1	1
P9975		2	3	4	7	7
Total		60	83	93	114	126
Total score (%)		38,22	52,87	59,24	72,61	80,25

Table 3. Number of correctly classified test cases in the top-k choice for each problem type.

The overall performance of the proposed algorithm ranges from 38.22% for the top-1 choice up to 80.25% for the top-5 choice. However, the performance varies significantly among the problem type categories. For example, for the problem type P9965 the corresponding score ranges from 0% for the top-1 choice up to 20% for the top-5 choice, whereas for the problem type P7695 the score for the top-1 choice is 64.86% and goes up to 100% for the top-5 choice. This is mainly due to insufficient training data to calculate

in a reliable way the conditional probabilities for the event codes of the Bayesian model. For example, in the case of problem type P9965 there are only two training cases, whereas for problem type P7695 there are 17 instances in the training dataset.

In addition, if the event codes for a problem type are not balanced between the training and test cases then in the likelihood term of Eq. (12) there will be more event codes with conditional probabilities close to zero that will make very unlikely the specific problem type to be the MAP estimate of the recommender. Another factor that needs further experimentation is related to the underline assumption that the event codes are independent, which in general may not be the case. To circumvent such a problem one would need to consider statistics for detecting correlations of event codes for each problem type category. This of course is an even harder problem if you consider the small number of available training data.

Further work is needed in the direction of defining adequate performance metrics, depending on the nature of the PHM problem at hand (Zhou et al., 2013). Metrics may take into account in different ways performance issues, such as false positives and false negatives in detection tasks, precision and recall in diagnosis tasks, as well as accuracy and confidence in prognostics. Adjusting the adopted metrics to the priorities of the problem at hand, for example by making it important to avoid false positives or false negatives, may also lead to the adoption of a vector of multi-objective, rather than a single scalar performance criterion.

5. CONCLUSION

A Bayesian method for maintenance action recommendation that was submitted to the PHM Data Challenge 2013 was presented. The method is characterized for its simplicity and the low computational resources it requires. The method incorporates prior information about the event codes for the targeted problem types. The prior information can also be improved after each test case is classified in a problem type. However, the method requires a balanced set of event codes for each problem type between training and test cases. Future work will try to circumvent this drawback as well as to address the task of detecting nuisance cases. A hybrid recommender that fuses the ability to detect nuisance cases with that of identifying the most likely problem type could lead to improved performance.

ACKNOWLEDGEMENT

This work is partly funded by the project Wireless Sensor Networks for Engineering Asset Life Cycle Optimal Management (WelCOM / 09SYN-71-856) of the Greek General Secretariat of Research and Technology in the framework of the National Strategic Reference Framework (NSRF) 2007-2013 programme for development. The authors would like to thank the organizers of the 2013 PHM

Data Challenge, Dustin Garvey and Neil H. Eklund, from GE Global Research for providing prompt evaluation feedback during the competition and sharing the ground truth data after the completion of the competition.

REFERENCES

- Eker, OF., Camci, F., and Jennions, IK., (2012), Major Challenges in Prognostics: Study on Benchmarking Prognostics Datasets, Proc. of PHM 2012, First European Conference of the Prognostics and Health Management Society 2012, 3-5 Jul 2012, Dresden, Germany.
- Elnahrawy E. and Nath B. (2004). Context-Aware Sensors. H. Karl, A. Willig, A. Wolisz (Eds.): EWSN 2004, LNCS 2920, pp. 77–93. doi: 10.1007/978-3-540-24606-0_6
- Karandikar, J. M., Abbas, A. & Schmitz, T. L. (2013). Remaining useful tool life predictions in turning using Bayesian inference. *International Journal of Prognostics and Health Management*, vol. 4 (2) 025, pages 11.
- Koller, D., & Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, Cambridge, Massachusetts, USA.
- Mehta, P., Werner, A., and Mears, L. (2013), Condition based maintenance-systems integration and intelligence using Bayesian classification and sensor fusion, *Journal of Intelligent Manufacturing*, (in press), doi 10.1007/s10845-013-0787-1.
- Nectoux, P., Gouriveau, R., Medjaher, K., Ramasso, E., Morello, B., Zerhouni, N., and Varnier, C., (2012), PRONOSTIA: An Experimental Platform for Bearings Accelerated Life Test, IEEE International Conference on Prognostics and Health Management, IEEE PHM 2012, June, 2012, Denver, CO, USA.
- Saha, B. & Goebel, K. (2008). Uncertainty Management for Diagnostics and Prognostics of Batteries using Bayesian Techniques. *Proceedings of IEEE Aerospace Conference.*, March 1-8, Big Sky, MO. doi: 10.1109/AERO.2008.4526631
- Zhou, Y., Bo, J., Jie, Z., and Mingwei, G., (2013), Performance Metrics Assessment Method on Aircraft Prognostics and Health Management, Proc. of ICMTMA 2013, The Fifth International Conference on Measuring Technology and Mechatronics, 16-17 Jan 2013, Hong, Kong. pp. 799-802, doi: 10.1109/ICMTMA.2013.199

BIOGRAPHIES

Vassilis Katsouros received his M.Eng. degree in Electrical and Computer Engineering from the National Technical

University of Athens (NTUA), Greece, in 1992. In 1993 he obtained the M.Sc. in Communications and Signal Processing from Imperial College of Science Technology and Medicine of the University of London, UK. From 1993 until 1997 he worked as research assistant at the Interdisciplinary Research Centre for Process Systems Engineering (IRC-PSE) of Imperial College. In 1997 received his Ph.D. degree from Imperial College. Since 1998 he has been working at the Institute for Language and Speech Processing (ILSP) of the Athena Research and Innovation Centre, Greece where he is currently a Senior Researcher and a member of the Scientific Advisory Board. He has been involved in a numerous research projects at a national and European level. His research interests involve signal processing, probability and statistics, game theory, image processing and pattern recognition. He is a member of IEEE, ACM and Technical Chamber of Greece.

Vassilis Papavassiliou received his M.Eng. degree in Electrical and Computer Engineering from the National Technical University of Athens (NTUA), Greece, in 1998. In 2000 he obtained the M.Sc. in Language Technology from National and Kapodistrian University of Athens and NTUA, Greece and in 2010 received his Ph.D. degree also from NTUA. Since 2000 he has been working at the Institute for Language and Speech Processing (ILSP) of the Athena Research and Innovation Centre, Greece. He has been involved in a numerous research projects at a national and European level. His research interests involve document image analysis, handwriting recognition and text analytics.

Christos Emmanouilidis holds a Diploma in Electrical Engineering from the Aristotle University of Thessaloniki, Greece (1992) a M.Sc. from the School of Engineering, University of Durham, UK (1998) and a PhD from the School of Computing and Engineering Technology, University of Sunderland, UK (2002). He is currently a Senior Researcher at the Institute for Language and Speech Processing (ILSP) of the Athena Research and Innovation Centre, Greece. He has been involved for more than 15 years in numerous research projects in areas such as Condition Monitoring & Engineering Asset Management, Automated NDT, Industrial Automation and Applied Intelligent Systems. He is a Senior Member of the IEEE, a Founding Fellow of the International Society of Engineering Asset Management (ISEAM), Founding Board Member of the Hellenic Maintenance Society (HMS), and a member of the European Federation of National Maintenance Societies (EFNMS) European Asset Management Committee (EAMC) and the IFIP WG5.7 ‘Advances in Production Management Systems’.