Random Forests Identification of Gas Turbine Faults

Manolis Maragoudakis, Euripides Loukis and Panagiotis-Prodromos Pantelides
University of the Aegean, Samos, Greece
{mmarag,eloukis,pantelides}@aegean.gr

Abstract

In the present paper, Random Forests are used in a critical and at the same time non trivial problem concerning the diagnosis of Gas Turbine blading faults, portraying promising results. Random forests-based fault diagnosis is treated as a Pattern Recognition problem, based on measurements and feature selection. Two different types of inserting randomness to the trees are studied, based on different theoretical assumptions. The classifier is compared against other Machine Learning algorithms such as Neural Networks, Classification and Regression Trees, Naive Bayes and K-Nearest Neighbor. The performance of the prediction model reaches a level of 97% in terms of precision and recall, improving the existing state-of-the-art levels achieved by Neural Networks by a factor of 1.5%-2%. Furthermore, emphasis is given on the pre-processing phase, where feature selection and outliers identification is carried out, in order to provide the basis of a high performance automated diagnostic system. The conclusions derived are of more general interest and applicability.

1. Introduction

Development of effective Gas Turbine Condition Monitoring and Fault Diagnosis methods has been the target of considerable research in recent years. This is due to the high cost, sensitivity and importance of these engines for most industrial companies. Most of this research is directed towards the diagnosis of Gas Turbine blading faults, because of the catastrophic consequences that these faults can have, if they are not diagnosed in time. Even very small blading faults can very rapidly grow and result to huge destructions ([1], [2], [3]).

Blading faults diagnosis is regarded to be a very difficult problem, because of the high levels of noise in all relevant measurements and the high interaction between the numerous Gas Turbine blading rows.

Therefore, it is very important to take advantage of the processing power of modern computers, in order to provide a fast and reliable engine condition diagnosis from available measurements and to develop the highest possible level of intelligence and assistance to the operation and maintenance personnel.

The Gas Turbine Blading Fault Diagnosis problem was originally addressed in [4] and [5], based on classical pattern recognition methods. In the present paper, Random Forests, an ensemble classification methodology with promising characteristics, is applied for the first time on the task at hand. Applying two different types of randomness insertion into the individual trees of a forest, we evaluate which performs better, as opposed to results obtained by other classical Machine Learning algorithms, such as Neural networks, Classification and Regression Trees (CART), Naive Bayes, and K-nearest neighbor (KNN). As regards to Neural networks, researchers mention that they portray the best results among the other methodologies, at a level of 95%-96% [6]. Our contribution to the domain, is the introduction of an ensemble classifier, namely Random Forests, which outperforms all previous attempts to Gas Turbine Blading Fault Diagnosis. Furthermore, Random Forests can provide some insight on the inter-relationships between input features, unlike Neural nets, thus directing domain experts at selecting which measurement tools to use in real world applications.

The outline of the paper is as follows: in Section 2, the Gas Turbine Blading Fault Diagnosis problem is described, as well as the specific faults to be diagnosed and the type of instrumentation and measurements used. In this section, some preprocessing issues are also introduced, regarding feature selection and outlier identification, from which the corresponding pattern (feature vector) is calculated. In Section 3, the implementation of the Random Forests approach to the classification problem is described. In Section 4, the

experimental results are discussed, followed by some concluding remarks.

2. Problem and data description

The present work is based on data acquired from dynamic measurements on an industrial Gas Turbine into which different faults were artificially introduced. During the experimental phase four categories of measurements were performed simultaneously:

1. Unsteady internal wall pressure (using fast response transducers P2 to P5).
2. Casing vibration (using accelerometers A1 to A6 mounted to the outside compressor casing).
3. Shaft displacement at compressor bearings (using transducer B).
4. Sound pressure levels (using double layer microphone M).

A schematic representation of the Gas Turbine schema, illustrating the measuring instruments' arrangement is depicted in Figure 1.

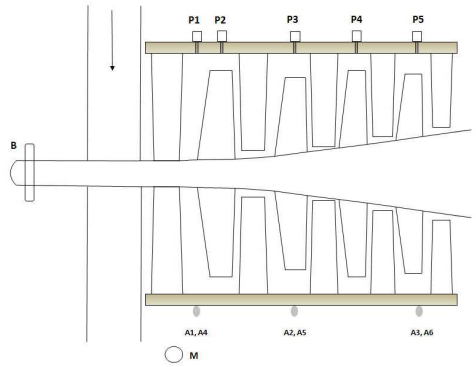

Figure 1. *Arrangement of the measuring instruments. Positioning directives: A1 and A4 are at the same position, only the latter is rotated by 90 degrees. In a similar manner, A2 with A5 and A3 with A6.*

Five experiments were performed, testing the datum healthy engine and a similar engine with the following four typical small (but quite rapidly growing, as mentioned in the introductory section) and also not straightforwardly diagnosable faults:

1. Fault1: Rotor fouling.
2. Fault2: Individual rotor blade fouling.
3. Fault3: Individual rotor blade twisted (by appr. 8 degreed).
4. Fault4: Stator blade restaggering.

Tests were performed at four different engine loads (full load, half load, quarter load and no load), both for the healthy engine as well as for the above four faults. At each load, four series of time domain data were acquired for each instrument (two series in each of the two sampling frequencies, l = 13 kHz and m = 32 kHz). The fault signatures were initially calculated in the form of spectral difference patterns, defined by the following expression:

$$P(f) = 20[\log(sp(f)) - \log(sph(f))]$$

where *P(f)* is the spectral difference pattern, which is a function of frequency *f*, *sp(f)* is the power spectrum of the signal of the measuring instrument from a faulty engine, and *sph(f)* is the signal spectrum from a healthy engine at the same load, sampling frequency and measurement series. Also, the most useful diagnostic information is contained at the harmonics of the shaft rotational frequency. This led to filtering out the values of *P(f)* at frequencies other than the shaft rotational frequency harmonics. The resulting pattern from this filtering, *Pr(f)*, is referred to as reduced spectral difference pattern (and for simplicity *'pattern'* in the following), and is given by the following equation:

$$\Pr(f) = P(f) * H(f)$$

where *H(f)=1*, if f is a rotational harmonic, and *H(f)=0,* for all other frequencies.

Patterns were calculated for frequencies up to the 27-th harmonic of the shaft rotational frequency, i.e. patterns belong to a 27-dimensional space [7]. An example of the pattern calculation procedure described above is shown in the following figure for power spectra of unsteady pressure transducer P2.

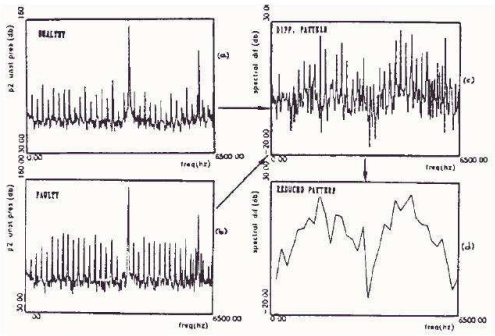

Figure 2. *Pattern calculation procedure for power spectra of unsteady pressure transducer P2.*

2.1. Goal of the study

As we described above, for the present Gas Turbine Blading Fault Diagnosis problem, twelve (12) different

measuring instruments were installed. Our goal is to check whether we are able to use Data Mining techniques as a diagnostic tool, in order to reduce the number of these instruments to one or two. In this manner we can achieve an important cost reduction, taking the fact that measurement series of this kind are costly and the maintenance personnel have to be specialized as well into consideration.

Nevertheless, if we are able to forecast a potential damage, we can deter machine's downfall, which entails substantial costs for any enterprise. The objective is to find a model and an instrument designing such that can foretell quite reliably a Gas Turbine's fault condition.

2.2. Data Description

As mentioned before, 12 different measuring instruments were used and measurements were taken for every possible combination between engine's 5 operational conditions (healthy engine and 4 faulty conditions), 4 different engine loads (full load, half load, quarter load and no load) and 2 sampling frequencies (low and high). To be more precise, regarding engine's healthy condition, measurements have been taken for every combination between the engine load and sampling frequency (total 8 different combinations). Especially in engine's faulty condition there's been one more measurement series for all the above combinations. Consequently, for every instrument we have aggregately 72 different measurements: 8 healthy engine's measurements and 64 faulty engine's measurements. For every instrument, each and every one of the above measurements consists of 27 values that are forms of the spectral difference of the first 27 harmonics of rotor's shaft rotational frequency. So, if we would like to present the entirety of data in a data base then this would be composed of 864 instances described by 27 distinct attributes, corresponding to the 27 harmonics that were mentioned above.

2.3. Pre-processing

Since a large number of input sensors was used, feature selection and outlier identification issues had to be confronted, in order for the classification model to be built in a robust and effective manner. In order to perform feature selection, we estimated the importance of each variable using the Gini index and sorted them in descending order (from the most important to the least important value). The feature importance graph provided important information regarding the significance of each variable to the classification process. By using correlation tests, these results were verified. As a last pre-processing step, removal of noisy data points was carried out, using statistical techniques [7].

3. Random forests

Despite the fact that Random Forests ([8], [9]) have been quite successful in classification and regression tasks, to the best of our knowledge, there has been no research in using the afore-mentioned algorithm for Gas Turbine Fault Diagnosis.

Nowadays, numerous attempts in constructing ensemble of classifiers towards increasing the performance of the task at hand have been introduced ([10], [11], [12]). A plethora of them has portrayed promising results as regards to classification approaches. Examples of such techniques are Adaboost, Bagging and Random Forests [13]. Random Forests are a combination of tree classifiers such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. The generalization error of a forest of tree classifiers depends on the strength of the individual trees within the forest and their inter-correlation. Using a random selection of features in order to split each node yields output error rates that compare equally to Adaboost, yet they are more robust with respect to noise. While traditional tree algorithms spend a lot of time choosing how to split at a node, Random Forests perform this task with little computational effort. Compared with Adaboost, Random Forests portray the following characteristics:

- the accuracy is as good as Adaboost and sometimes better.
- they are relatively robust to outliers and noise.
- they are faster than bagging or boosting.
- they provide useful internal estimates of error, strength, correlation and variable importance.
- they are simple and easily parallelized.

A Random Forest multi-way classifier $\Theta(x)$ consists of a number of trees, with each tree grown using some form of randomization, where x is an input instance [14]. The leaf nodes of each tree are labeled by estimates of the posterior distribution over the data class labels. Each internal node contains a test that best splits the space of data to be classified. A new, unseen instance is classified by sending it down every tree and aggregating the reached leaf distributions. The process is described in figure 3. Each tree is grown as follows:

- If the number of cases in the training set is N, sample N cases at random but with replacement, from the original data. This sample will be the training set for growing the tree.
- If there are M input variables, a number m<<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.
- Each tree is grown to the largest extent possible. Therefore, no pruning is applied.

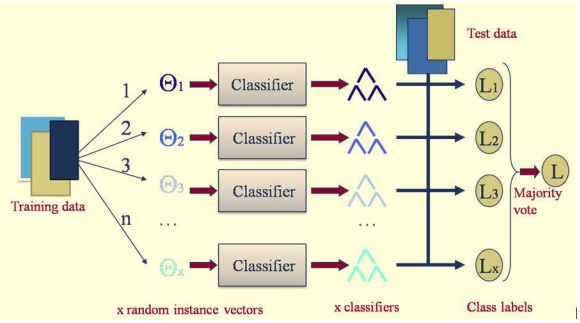

Figure 3. *Hierarchical decomposition of a Random Forests classifier on a data set*

As regards to the overall error rate of the Random Forests, this is affected by two different factors:

1. Tree *inter-correlation*. Highly correlated trees result in high error rate.
2. *Robustness* (strength) of each individual tree within the forest. Higher strength results in lower error rates.

Upon completion of the tree construction step, the set of data are run down the tree, and proximity values are computed for each pair of cases. If two cases occupy the same terminal tree node, their proximity is augmented by one. At the end of the run, proximities are normalized, divided by the number of trees.

In order to make the classification process more formal, suppose that the joint classifier $\Theta(x)$ contains x individual classifiers $\Theta_1(x)$, $\Theta_2(x)$,...,$\Theta_x(x)$. Let us also assume that each data instance is a pair (x,y), where x denotes the input attributes, taken from a set Ai, i=1,...,M and y symbolizes the set of class labels L_j, j=1,...,c (c is the number of class values). For reasons of simplicity, the correct class will be denoted as y, without any indices. Each discrete attribute A_i takes values from a set V_i, i=1 to m_i (m_i is the number of values attribute A_i has). Finally, the probability that an attribute A_i has value v_k is denoted by $p(v_i,k)$, the probability of a class value y_j is denoted by $p(y_j)$ and the probability of an instance with attribute A_i having value v_k and class label y_j is symbolized by $p(y_j|v_i,k)$.

Each training example is picked up from a set of N instances at random with replacement. By this procedure, called bootstrap replication, a pool of 36.8% of the training examples are not used for the tree construction phase. These out-of-bag (oob) instances allow for computing the degree of strength and correlation of the forest structure. Suppose that $O_k(x)$ is the set of oob instances of classifier $\Theta_k(x)$. Furthermore, let $Q(x, y_j)$ denote the subset of oob samples which were voted to have class y_j at input example x. An estimate of $p(\Theta(x) = y_j)$ is given by the following equation:

$$Q(x, y_j) = \frac{\sum_{k=1}^{K} I(\Theta_k(x) = y_j; (x, y) \in O_k)}{\sum_{k=1}^{K} I(\Theta_k(x); (x, y) \in O_k)}$$

where $I(\cdot)$ is the indicator function.

The margin function which measures the extent to which the average vote for the right class y exceeds the average vote for any other class labels is computed by:

$$\text{margin}(x, y) = P(\Theta(x) = y) - \max_{j=1, j \neq y}^{c} (P(\Theta(x) = y_j)$$

Since strength is defined as the expected margin, it is computed as the average over the training set:

$$s = \frac{1}{n} \sum_{i=1}^{n} (Q(x_i, y) - \max_{j=1, j \neq y}^{c} Q(x_i, y_j))$$

The average correlation is given by the variance of the margin over the square of the standard deviation of the forest: $\bar{p} = \frac{Var(\text{margin})}{\sigma(\Theta())^2}$, is estimated for every input example x in the training set Q(x, y_j).

We used unpruned decision trees as base classifiers and introduced two different additional methods for randomness to the trees. Following Breiman's approach, we utilized the Random Input Forests and the Random Combination Forests algorithms. Nevertheless, for both methods, the evaluation metric on which tree nodes are chosen to split is the Gini index, taken from the CART algorithm. Other similar metrics presented by researchers are Gain ratio [15], MDL [16] and Relief-F [17]. The formula of the Gini index is as follows [18]:

$$Gini(A_i) = -\sum_{i=1}^{c} p(y_i)^2 - \sum_{j=1}^{m_i} p(v_{i,j}) - \sum_{j=1}^{c} p(y_i / v_{i,j})^2$$

3.1. Random input forests

The simplest random forest with random features is formed by selecting at random, at each node, a small group of input variables to split on. Grow the tree using CART methodology to maximum size and do not prune. The process is tabulated below:

For building K trees:

- Build each tree by:
  - Selecting, at random, at each node a small set of features (F) to split on (given M features). Common values of F are:
    - 1. F=1.
    - 2. F=log2(M) + 1.
  - For each node split on the best of this subset (using oob instances).
  - Grow tree to full length.

3.2. Random combination forests

This approach consists of defining more features by taking random linear combinations of a number of the input variables. That is, a feature is generated by specifying L, the number of variables to be combined. At a given node, L variables are randomly selected and added together with coefficients that are uniform random numbers on [-1,1]. F linear combinations are generated, and then a search is made over these for the best split. The complete procedure is as follows:

For building K trees:

- Build each tree by:
  - Create F random linear sums of L variables:

$$A_f = \sum_{i=1}^{L} b_{fj} x_i$$

, where b_{fj} = uniform random number on [-1,+1]

  - At each node split on the best of these linear boundaries.
  - Grow tree to full length.

4. Experimental results

We applied the two versions of Random Forests (Random Input (RI) Forests and Random Combination (RC) Forests) on the Gas Turbine data set, using oob estimates. As for evaluation metric, we considered per-class precision and recall. Accuracy in some domains, such as the one at hand, is not actually a good metric due to the fact that a classifier may achieve high accuracy by simply always predicting the non faulty class. This problem particularly appears in the present task, where, from more than 2/5 of the data set contained the afore-mentioned class. A set of well-known machine learning techniques have constituted the benchmark to which our results have been compared: Multilayer Perceptron Neural Networks, Naive Bayes, Classification and Regression Trees (CART), and k-Nearest Neighbor (kNN) instance based learning. Cross-validation was performed with kNN in order to determine the best k.

As regards to the Random Forests implementation, the best results were obtained by using 500 trees and 6 features. The evaluation outcomes are depicted in the following two figures (F1 to F4 denotes the fault categories and OK denotes the non faulty state)

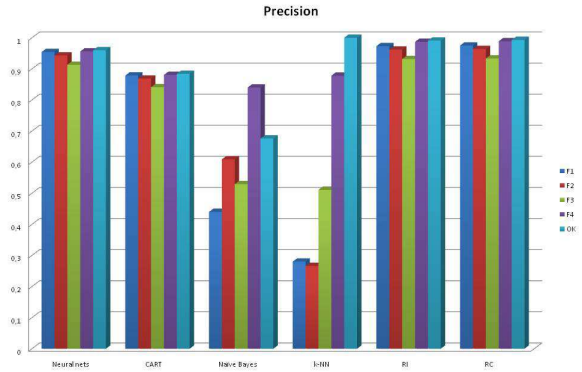

Figure 4. *Precision metric for all methodologies.*

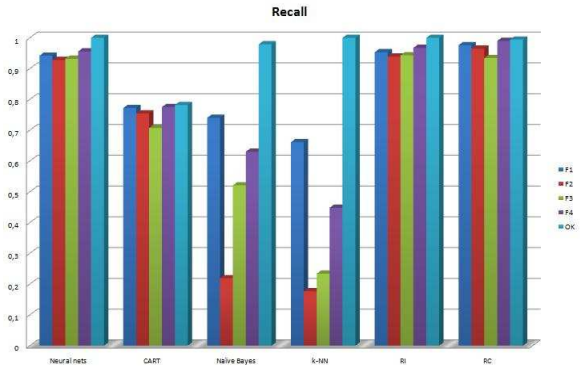

Figure 5. *Recall metric for all methodologies.*

Both of the Random Forests implementations outperform all other algorithms. RC is slightly better that RI, however the increase is very small, so that we could claim that they perform similarly.

Due to lack of space, we only provide the margin curve for the most effective algorithm (RC) in figure 6. The margin curve prints the cumulative frequency of the difference of the actual class probability and the highest probability predicted for the other classes As can been observed, the majority of instances are

correctly classified by the Random Forests model, since they are located near the area of probability one (the right part of the graph).

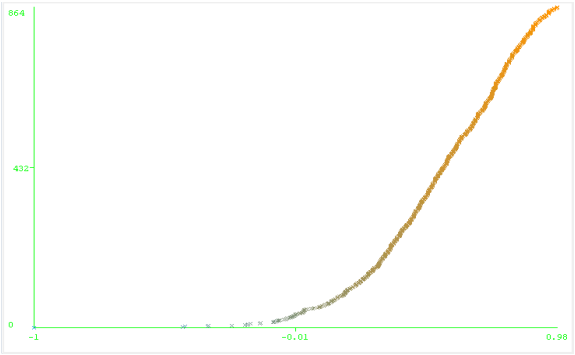

Figure 6. *The margin curve as extracted for the RC algorithm.*

5. Conclusion

We have shown that an ensemble method such as Random Forests is well suited for the task of predicting Blading faults on a Gas Turbine. Results have indicated that, regardless of the method of injecting randomness to the trees of the forest, that algorithm outperforms all previous approaches and presents state-of-the-art outcomes in terms of precision and recall. More specifically, both Random Input Forests and Random Combination Forests appeared to be more accurate than Naive Bayesian, Neural Networks, Classification and Regression Trees and k-Nearest Neighbor classifiers. Furthermore, the structure of the forests can provide essential feedback to the domain experts, as regards to the most effective (both in accuracy and cost figures) number of measuring units required for the creation of an automated diagnosis framework with desirable characteristics.

The reduction in measuring units is beneficial since there is a significant decrease in cost. Nevertheless, we are of the belief that such ensemble methods like the one at hand can be applied to other domains with similar robust behavior.

6. References

[1] E. Loukis, P. Wetta, K. Mathioudakis, A. Papathanasiou, K. Papailiou, Combination of Different Unsteady Quantity Measurements for Gas Turbine Blade Fault Diagnosis, 36th ASME International Gas Turbine and Aeroengine Congress, Orlando, 1991, ASME paper 91- GT-201.

[2] E. Loukis, Contribution to Gas Turbine Fault Diagnosis Using Methods of Fast Response Measurement Analysis, Doctoral Thesis, Athens, National Technical University of Athens, 1993.

[3] G. Merrington, O. K. Kwon, G. Godwin, B. Carlsson, Fault Detection and Diagnosis in Gas Turbines, ASME Journal of Engineering for Gas Turbines and Power, 113, 1991, 11-19.

[4] E. Loukis, K. Mathioudakis, K. Papailiou, A procedure for Automated Gas Turbine Blade Fault Identification Based on Spectral Pattern Analysis, Journal of Engineering for Gas Turbines and Power, 114, 1992, 201-208.

[5] E. Loukis, K. Mathioudakis, K. Papailiou, Optimizing Automated Gas Turbine Fault Detection Using Statistical Pattern Recognition, Journal of Engineering for Gas Turbines and Power, 116, 1994, 165-171.

[6] Roberto Battiti and Anna Maria Colla, Democracy in Neural Nets: Voting Schemes for Classification, Neural Networks, 7, 4, 1994, 691-707.

[7] A.D.Pouliezos and G.S.Stavrakakis, Real Time Fault Monitoring of Industrial Processes, Dordrecht, Kluwer Academic Publishers, 1994.

[8] Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. Classification and regression trees. Wadsworth Inc., Belmont, California, 1984.

[9] Leo Breiman. Bagging predictors. Machine Learning Journal, 26(2):123140, 1996.

[10] Leo Breiman. Random forests.Machine Learning Journal, 45:532, 2001.

[11] Yoav Freund and Robert E. Shapire. Experiments with a new boosting algorithm. In Lorenza Saitta, editor, Machine Learning: Proceedings of the Thirteenth International Conference (ICML96). Morgan Kaufmann, 1996.

[12] Y. Amit and D. Geman. 1997. Shape quantization and recognition with randomized trees. Neural Computation, (9):15451588.

[13] T.K. Ho. 1998. The random subspace method for constructing decision forests. IEEE Trans. on Pattern Analysis and Machine Intelligence, 20(8):832844.

[14] Igor Kononenko. Estimating attributes: analysis and extensions of Relief. In Luc De Raedt and Francesco Bergadano, editors, Machine Learning: ECML-94, pp. 171182. Springer Verlag, Berlin, 1994.

[15] J. Ross Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco, 1993.

[16] Igor Kononenko. On biases in estimating multi-valued attributes. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI95), pp. 10341040. Morgan Kaufmann, 1995.

[17] Breiman, L.: Looking Inside the Black Box, Wald Lecture II, Department of Statistics, California University, 2002.

[18] Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. Classification and regression trees. Wadsworth Inc., Belmont, California, 1984.