Towards Relation Discovery for Diagnostics

Rune Prytz  
Volvo Technology  
Götavärssgatan 10  
405 08 Göteborg  
Rune.Prytz@volvo.com

Slawomir Nowaczyk  
Halmstad University  
Box 823  
301 18 Halmstad  
Slawomir.Nowaczyk@hh.se

Stefan Byttner  
Halmstad University  
Box 823  
301 18 Halmstad  
Stefan.Byttner@hh.se

ABSTRACT

It is difficult to implement predictive maintenance in the automotive industry as it looks today, since the sensor capabilities and engineering effort available for diagnostic purposes is limited. It is, in practice, impossible to develop diagnostic algorithms capable of detecting many different kinds of faults that would be applicable to a wide range of vehicle configurations and usage patterns.

However, it is now becoming feasible to obtain and analyse on-board data on vehicles as they are being used. It makes automatic data-mining methods an attractive alternative, since they are capable of adapting themselves to specific vehicle configurations and usage. In order to be useful, though, such methods need to be able to detect interesting relations between large number of available signal.

This paper presents an unsupervised method for discovering useful relations between measured signals in a Volvo truck, both during normal operations and when a fault has occurred. The interesting relationships are found in a two-step procedure. In the first step, we identify a set of “good” models, by establishing an MSE threshold over the complete data set. In the second step, we estimate model parameters over time, in order to capture the dynamic behaviour of the system. We use two different approaches here, the LASSO method and the Recursive Least Squares filter. The usefulness of obtained relations is then evaluated using supervised learning to separate different classes of faults.

Categories and Subject Descriptors

I.5.4 [Pattern recognition]: Applications — Signal Processing

General Terms

Algorithms and Reliability

Keywords

Fault detection, Vehicle diagnostics, Machine learning

1. INTRODUCTION

Mechatronic systems of today are typically associated with a high software and system complexity, making it a challenging task to both develop and, especially, maintain those systems. For commercial ground vehicle operators (such as bus and truck fleet owners), the maintenance strategy is typically reactive, meaning that a fault is fixed only after it has occurred. In the vehicle industry it is difficult to move towards predictive maintenance (i.e. telling that there is a need for maintenance before something breaks down) because of limited budget for on-board sensors and the amount of engineering time it takes to develop algorithms that can handle different kinds of faults, but also work in a wide range of vehicle configurations and for many different types of operation under varying environment conditions.

If the current trend of increasing number of components in vehicles continues (alsongside with requirements on component reliability and efficiency), then the only solution will be to move towards automated data analysis to cope with increasing costs. At the same time, with the introduction of low-cost wireless communication, it is now possible to do data-mining on-board real vehicles as they are being used.

This paper presents an approach that allows discovery of relations between various signals that available on the internal vehicle network. It is based on the assumption that while it is difficult to detect faults by looking at signals (such as road speed) in isolation, the interrelations of connected signals are more likely to be indicative of abnormal conditions.

One requirement for our approach is to be able to perform relation discovery in a fully unsupervised way. This is important since, while an engineer may be able to predict a large number of “good” relations between various signals, her knowledge will in most cases be global, i.e. general enough to hold for all or almost all vehicles. An automated system, on the other hand, can be more tightly coupled to the specifics of a particular fleet of vehicles, either by geographic region, vehicle configurations or type of operation — for example, it is likely that some of the signal relations that hold in Alaska do not hold in Mexico, or that some of the relations that are useful for detecting faults in long-haul trucks would be inadequate for delivery trucks. It is not feasible to develop specialised diagnostic algorithms for each of those cases, unless fault detection is done in an automatic way.

This paper is organised as follows. In the following section we briefly highlight some of the related research ideas. After that, in section 3, we describe the data we have been working with. Section 4 presents our approach towards discovering relations, in three steps: data preprocessing, signal...
selection and model parameter estimation. Finally, we evaluate obtained results in section 5 using supervised learning, and we close with some conclusions in section 6.

2. RELATED RESEARCH

Automated data mining for vehicle applications has previously been the topic of several papers. An early paper by Kargupta et al. [4] shows a system architecture for distributed data-mining in vehicles, and discusses the challenges in automating vehicle data analysis. In Vachkov [6], the benefits of being able to do cross-fleet analysis (comparing properties of different vehicles) is shown to benefit root-cause analysis for pre-production diagnostics. In Byttner et al. [1], a method called COSMO is proposed for distributed search of “interesting relations” (e.g. strong linear correlations that hold for long periods of time) among on-board signals in a fleet of vehicles. The interesting relations can then be monitored over time to enable e.g. deviation detection in specific components. A method based on a similar concept of monitoring correlations (but for a single vehicle instead of a fleet) is shown in D’Silva [2]. In Vachkov [6], the neural gas algorithm is used to model interesting relations for diagnostic of hydraulic excavators. Contrary to our work, however, both the papers by D’Silva and Vachkov assume that the signals which contain the interesting relations are known a priori. In [5], a method for monitoring relations between signals in aircraft engines is presented. Relations are compared across a fleet of planes and flights. Unlike us, however, they focus on discovering relationships that are later evaluated by domain experts.

3. DESCRIPTION OF DATA

Measurements have been done on a Volvo VN780 truck with a D12D diesel engine. We have analysed a total of 14 driving runs, four of which were performed under normal operating conditions and the other ten with various faults introduced. The truck was equipped with a logging system which collected data from the internal vehicle network as well as from a set of extra sensors. In total, 21 signals were recorded with a sampling frequency of 1 Hz. Each driving run was approximately four hours in length, under a variety of driving conditions.

We have specifically targeted the air-intake system, since it is prone to wear and needs regular maintenance during the lifetime of a vehicle. Four different faults have been injected into the truck. The first two were clogging of air filter (AF) and grill. AF change and grill cleaning are routine maintenance operations that are performed regularly. The third fault was charge air cooler (CAC) leak. Such leaks occur in the joints between CAC and connecting air pipes and are considered faults that are hard to find manually. Finally, exhaust pipe was partially congested, which is a rather uncommon fault.

4. RELATION DISCOVERY

The method we use for discovering relations consists of three steps. We start with data preprocessing and remove influence of ambient conditions. Then, we proceed to choose the most interesting signals to model, as well as which signals should be used to model them. Finally, we estimate model parameters. In this last step we use two different approaches, the LASSO (Least Absolute Shrinkage and Selection Operator) method and RLS (Recursive Least Squares) method. In this work we are using MSE as the main criterion for determining which relations are interesting.

4.1 Preprocessing

First, we have performed normalisation of all signals, and removed obvious outliers. Then, available signals were divided into system and ambient signals (in the natural way, for example engine speed is a system signal while ambient temp and altitude are ambient signals).

In order to increase the SNR of signals, we begin by filtering out the effects of ambient conditions on the system, using a (slightly simplified) procedure introduced in [5]. Namely, we attempt to model each system signal $y_k$ as a linear combination of all ambient signals:

$$\Theta_k = \arg\min_{\Theta \in \mathbb{R}^s} \left( \sum_{t=1}^n (y_k(t) - \Theta^T \varphi_{amb}(t))^2 \right) \quad (1)$$

$$y_k^{norm}(t) = y_k(t) - \Theta_k^T \varphi_{amb}(t) \quad (2)$$

where $s$ is number of ambient signals, $y_k(t)$ is the value of a system signal $k$ at time $t$, $\Theta_k$ is a vector of parameter estimates for the model of $y_k$ and $\varphi_{amb}$ is the regressor for the model of $y_k$ (i.e. a vector of values of ambient signals at time $t$). Intuitively, $y_k^{norm}$ is this part of signal $y_k$ that cannot be explained by external conditions.

Figure 1: Signal normalisation

Figure 1 above illustrates how the intake air temperature is affected by the ambient conditions (mainly ambient air temperature). After ambient filtering, the signal has significantly less variance.

4.2 Signal selection

The next step is to find out which relations between signals are interesting. We perform this step in two stages. In the first stage we attempt to model each system signal using all other systems signals as regressors:

$$\Psi_k = \arg\min_{\Psi \in \mathbb{R}^{s-1}} \left( \sum_{t=1}^n (y_k(t) - \Psi^T \varphi_k(t))^2 \right), \sum_{t=0}^{n-1} ||\Psi_k|| < C_k \quad (3)$$
where $s$ is number of system signals, $\Psi_k$ is a vector of parameter estimates for the model of $y_k$ and $\varphi_k$ is the regressor for $y_k$ (i.e. the set of all other system signals). The energy constraint $C_k$ provides an upper bound on the sum of absolute values of all parameters for $y_k$. We gradually increase its value, performing a cross-validation test after each run. Initially, the mean squared error of the model keeps decreasing, but at some point it begins to increase, as seen in figure 2. The lasso constraint forces small parameters representing insignificant relations to go to zero and significant relations to be prioritised, resulting in models with less non-zero parameters than standard least squares approaches.

This approach allows an estimator to easily adapt to changing models (for example, it is easy to imagine that some relations look differently when truck is going downhill than when it is going uphill). On the other hand, when there are two (or more) different models that are similarly plausible, LASSO estimator is likely to oscillate between them in a nearly random way.

A second method is a Recursive Least Square filter [3], which recursively calculates the estimates over a sliding window defined by the forgetting factor. It aims to minimise a weighted linear least squares cost function, thus exhibiting very fast convergence at the cost of poor tracking performance (i.e. when the “true relation” to be estimated changes, it takes a long time for RLS to catch up with this change).

$$P(0) = \delta_{init}^{-1}I \quad \Theta(0) = \Theta_{init} \quad (4)$$

$$e(n) = g(n) - \Theta^T(n-1)\varphi(n) \quad (5)$$

$$g(n) = \frac{P(n-1)\varphi(n)}{\lambda + \varphi^T(n)P(n-1)\varphi(n)} \quad (6)$$

$$P(n) = \lambda^{-1}P(n-1) - g(n)\varphi^T(n)\lambda^{-1}P(n-1) \quad (7)$$

$$\Theta(n) = \Theta(n-1) + e(n)g(n) \quad (8)$$

The estimates from all RLS-estimators are collected into an array $W(t) = [\Theta(1) \cdots \Theta(t)]$, since each new sample from the system results in new, updated estimates.

Using the LASSO method, we obtain one set of model parameters for each time slice. With RLS, we get significantly more model parameters, but this data is more interdependent. While model parameters from LASSO method are all calculated from different parts of input time series, RLS models are all evolutions of the same predecessor.

Figures 3 and 4 show parameters used for modelling the signal fuel inst. The X and Y axis each represents one of the model parameters (i.e. dimensions in the input space of the classifier, as explained in the following section). The dots in the figures each correspond to a single estimate from the RLS estimator and from the LASSO estimator, respectively. As can be seen, our method has autonomously discovered that fuel inst (instantaneous fuel consumption) can be approximated using cac in p (charge air cooler input pressure) and in manif t (input manifold temperature). In other words, there exists a relation “fuel inst = $P_1* cac in p + P_2*in manif t$”.

The actual values of $P_1$ and $P_2$ parameters, of course, depend on the exact values of the signals in question, but as can be seen in figures 3 and 4, they show an interesting regularities. There are some differences between the two methods, but the general pattern is the same. It appears that some of the faults (in particular, clogged air filter) can be quite easily differentiated from normal operation, but some faults (especially CAC leaks) are impossible to detect. This is mainly due to two reasons. Firstly, the injected CAC leaks were very small in comparison to what is considered a serious problem, and secondly, there are very few sensors located sufficiently close to the fault area.

In a similar fashion, figures 5 and 6 show the parameters corresponding to relations “eng load = $P_1* fuel inst$” and “fuel inst = $P_2* cac in p$”. There is no direct physical correspondence between those two parameters, but as can be seen, at least in the case of RLS method, this relation can still be useful for detecting some faults.
Overall, though, it is rather difficult to evaluate the quality of discovered relations. Some of them make sense from the domain expert point of view in the general sense, but actual parameter values tend to vary greatly between different time slices. Therefore, we have decided to use supervised learning in order to analyse how much useful information is there in those relations.

5. Evaluation

We have used three different classifiers: linear regression [8], support vector machine (SVM) [9] and random forest [7]. Each classifier has been used for multi-class classification within the model parameter space generated during the system monitoring step, from either LASSO or RLS estimators. In all cases, the input to the classifier is array $W$, which contains all the estimates for all models found over time.

Both the forgetting factor (for RLS) and the data slice size (for LASSO) are parameters for tuning. Larger slices and forgetting factor gives better signal to noise ratio and a more robust solution. On the other hand, they are less sensitive to faults that only appear under certain conditions. In our case, a partially clogged air filter will only have a visible effect if the engine is running at high power, since this is the only situation when a large air flow is required.

In order to visualise the behaviour of our unsupervised relation discovery method, the classification task were run a number of times with different time slices and forgetting factors. Figures 7 and 8 present the result of that experiment. It is easily seen that choosing too small forgetting factor for RLS is detrimental. On the other hand, the effect of choosing too small data slices is not visible, at least for reasonable window sizes.

In general, the random forest classifier outperforms both SVM and linear classifier by a pretty large margin. Besides that, RLS estimator appears to be a little better than the LASSO estimator, but the difference is not huge (it is not clear if this difference is worth the significantly higher computational complexity). An interesting observation is that the number of data slices does not have a big impact on the classification accuracy, but there is a definite sweet point for the forgetting factor at $0.001$.

As a final comment, the resulting classification error appears to be rather high, but it is important to take into account that this data set is a pretty difficult one. There is
a lot of different external influences that disturb the “normal” operation of a truck, and the low quality of many of the sensors result in high levels of noise in the data. Also, for the predictive maintenance needs, it is not necessary to achieve 100% or close accuracy — it is usually enough to detect faults some proportion of the time, since we are often more interested in following gradual wear rather than abrupt failures. The seemingly low overall classification accuracy can also be partially attributed to the lack of dedicated sensors: in particular, neither of the four faults we have analysed is currently being detected for in-production vehicles.

6. CONCLUSIONS
In this paper we have presented a method for automatic discovery of interesting relations between time series of vehicle signal data. We have evaluated it on the data logged from a Volvo truck, and we have shown that resulting models can be used for diagnostics using supervised learning. This is an important step towards a system that would be able to analyse on-board data on real vehicles and detect anomalies in an autonomous way.

This is very much work in progress and there are numerous directions to extend those results. An obvious thing is to look into ways of improving classification accuracy: we have used three well-known learning algorithms with defaults settings, but there is room for improvement both in the learning phase itself, as well as in the estimation of model parameters. We have implemented two methods (LASSO and RLS), but there are many other potential solutions. Also, we have identified advantages and flaws of both of those methods, so it would be interesting to look into possibility of developing some kind of hybrid approach.

It is also not quite clear if the supervised classification is the best way of evaluating usefulness of discovered relations. We intend to explore other possibilities.

Finally, all the data we have access to at the moment comes from a single vehicle. The major benefit of unsupervised relation discovery lies in the possibility of generalising knowledge across multiple similar vehicles in a larger fleet. We are currently in the process of gathering data from a number of trucks and buses, so in the near future we should be able to evaluate our approach in that setting. We expect this to be more challenging due to higher variance within the data, but at the same time a fleet-based approach will give us access to significantly more signal readings.

7. ACKNOWLEDGEMENTS
The presented work has been partially funded by grants from VINNOVA and from the Knowledge Foundation.

8. REFERENCES