

USING CASED-BASED REASONING DOMAIN KNOWLEDGE TO TRAIN A BACK PROPAGATION NEURAL NETWORK IN ORDER TO CLASSIFY GEAR FAULTS IN AN INDUSTRIAL ROBOT

Erik Olsson

School of Innovation, Design and Engineering,
Mälardalen University, P.O. Box 883, SE-721 23 Västerås, Sweden,
phone +46 21 10 73 35, fax +46 21 10 14 60

ABSTRACT

The classification performance of a back propagation neural network classifier highly depends on its training process. In this paper we use the domain knowledge stored in a Case-based reasoning system in order to train a back propagation neural network to classify gear faults in an industrial robot. Our approach is to compile domain knowledge from a Case-based reasoning system using attributes from previously stored cases. These attributes holds vital information usable in the training process. Our approach may be usable when a light-weight classifier is wanted due to e.g. lack of computing power or when only a part of the knowledge stored in the case base of a large Case-based reasoning system is needed. Further, no use of the usual sensor signal classification steps such as filtering and feature extraction are needed once the neural network classifier is successfully trained.

KEYWORDS

Case-Based Reasoning, Neural Network, Sound recordings, Fault classification

INTRODUCTION

In this paper we use the domain knowledge already stored in a Case-based reasoning (CBR) (Amodt and Plaza, 1994) system in order to train a back propagation neural network (NN) to classify gear faults in an industrial robot. Our approach may be usable when a simple classifier is wanted due to e.g. lack of computing power, ease of use or when only a part of the knowledge stored in the case base of a large CBR system is needed. CBR offers a method to implement intelligent diagnosis systems for real-world applications (Nilsson et al., 2003). Motivated by the doctrine that similar situations lead to similar outcomes, CBR is able to classify sensor signals based on experiences of past categorizations saved as cases in a case-base. This paper is based on a CBR system used to diagnose audible faults in industrial robots (Olsson et al., 2004) as mechanical fault in industrial robots often show their presence through abnormal acoustic signals. The system uses CBR and acoustic signals as a proposed solution of recognizing audible deviations in the sound. The sound is recorded by a microphone and compared with previously made recordings; similar cases are retrieved and a diagnosis of the robot can be made. The system uses three different steps in its classification process; pre-processing, feature identification and classification. The pre-processing process is responsible for filtering and removal of unwanted noise. In

the feature identification process, the system uses a two-pass model, first identifying features and then creating a vector with features. Features are extracted using methods such as FFT and wavelet analysis (Lee and White, 1998), (Lin, 2001). A feature in the case is a normalized peak value at a certain frequency and time offset. Once the features are identified, the system classifies the feature vector. The classification is based on previously classified measurements stored as cases in a case base. Cases are retrieved using a nearest neighbor function that calculates the Euclidian distance between the new case and the cases stored in the case library. A list with the k nearest neighbors is retrieved based on the distance calculations. When a new sound has been classified, the system learns by adding it as a new case to the case-base. At recent time, the system stores classified cases of recordings of gearboxes from 24 healthy and 6 faulty robots. We have used CBR domain knowledge from two of those cases in order to train a NN classifier to classify one type of gear fault. Our approach may be usable when only a small and simple classifier is wanted that might use only a part of the knowledge stored in a CBR system. Further, no use of the usual sensor signal classification steps involving filtering and feature extraction are needed. Once successfully trained, the neural network classifier can be directly applied on noisy sensor data and it will represent the part of the case-base used in its training process and it will respond accordingly e.g. it might act as a red/green light in response to its input, signaling a failed/normal gearbox. The paper is organised as follows: section 2 gives a formal description of the CBR system that is used as the source for domain knowledge. Section 3 presents the method used to extract this domain knowledge. Section 4 describes how to train a simple NN classifier using the extracted domain knowledge. Section 5 presents an evaluation of the classification performance of the neural network classifier and section 6 gives a brief conclusion of the paper.

THE CBR SYSTEM

The CBR system consists of the tuple (CB, sim) (Perner, 2007) where CB denotes its case-base and sim is a similarity function that classifies a case by searching for similar cases already processed and stored in the case base. A case base (CB) contains a sequence of n cases X_1, X_2, \dots, X_n . The cases are indexed in a flat hierarchy with $CB = (X_1, X_2, \dots, X_n)$. A case X is a triplet $(x, FV, class(FV))$ where x is the unprocessed sensor signal, FV is a feature vector containing m features describing the nature of the sound recording and $Class(FV)$ is the class of X . In our system, each feature F in FV is a triplet (A, t, f) describing a peak in sensor data x with amplitude A , frequency f and location offset at time t . Features are extracted from the sensor data by means of various methods such as FFT, wavelet analysis [5, 6] etc.

The class of X is determined by the similarity function

$$sim(FV_1, FV_2) = \sqrt{\sum_{i=1}^m (FV_{1i} - FV_{2i})^2}$$

measuring the Euclidian distance between two feature vectors FV_1 and FV_2 . The k nearest neighbours of X indexed by FV_1 are retrieved and the class of X is determined by the class of these neighbours.

EXTRACTING DOMAIN KNOWLEDGE

The domain knowledge for $class_i$ can be seen as the information stored in the cases contained in the cluster CB_i formed by all cases X_i having feature vector FV where $class(FV) = i$, and consequently, domain knowledge for $class_j$ can be seen as the information stored in the cases contained in the cluster CB_j formed by all cases X_j having feature vector FV where $class(FV) = j$ etc. Our approach is to use

cluster CB_i to train a stand-alone NN classifier to classify sensor data of $class_i$, and consequently, a cluster CB_j can be used to train the same stand-alone NN classifier to classify sensor data of $class_j$ etc. In this manner, explicit domain knowledge from a case base can be transferred and transformed into implicit domain knowledge inside a NN classifier.

We have used CBR domain knowledge in the form of time offsets from two cases X_i and X_j in order to train a two-layer back propagation neural network (Bishop, 1995). The cases contain sound recordings from a normal ($class_i$) and a broken ($class_j$) gearbox originating from two industrial robots. The output gear in the broken gearbox had a broken gear tooth generating impact sounds (Barber, 1992) whereas the normal gearbox did not generate any impact or any other abnormal sound whatsoever. Both sound recordings were contaminated with noise originating from the gearboxes themselves and from the noisy factory environment the robots was situated in. Case X_i and X_j are described as follows

$$X_i = \{x_i, FV(NULL), class(FV) = i\}$$

$$X_j = \{x_j, FV([A(0.5), t(3.1), f(50)], [A(0.4), t(5.1), f(50)])), class(FV) = j\}$$

Where case X_i contains no description of an impact or any other abnormal sound whatsoever and case X_j reveals two impulse sound peaks; $peak_1$ and $peak_2$, caused by a broken gear tooth on the output gear. $peak_1$ is located at time offset $t(3.1)$ with an amplitude of $A(0.5)$ and a frequency of $f(50)$. $peak_2$ is located at time offset $t(5.1)$ with an amplitude of $A(0.4)$ and a frequency of $f(50)$. Figure 1 depicts the unprocessed sound signal x_j as it is stored in case X_j and Figure 2 shows the same sound recording as depicted in figure 1 but filtered at frequency $f(50)$ in order to reveal impulse sound $peak_1$ and $peak_2$ caused by the broken gear tooth on the output gear.

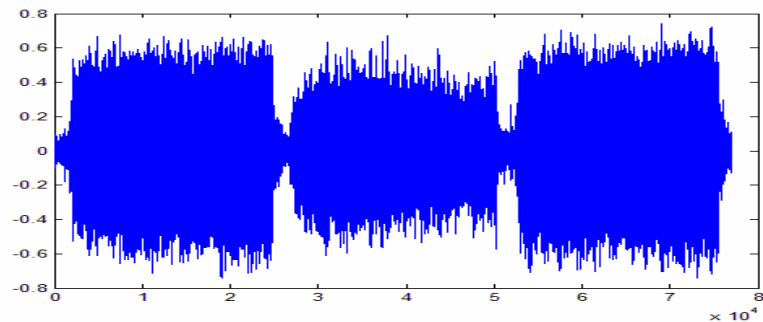


Figure 1 Unfiltered sound recording from the gearbox

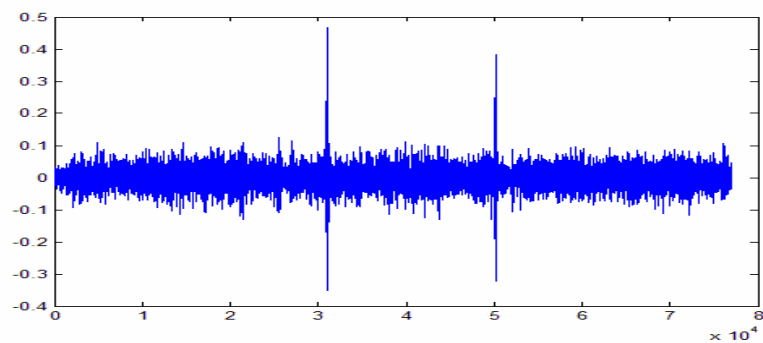


Figure 2 Impulse sound peaks hidden in the noisy sound recording

TRAINING A NEURAL NETWORK CLASSIFIER

A two-layer (Bishop, 1995) NN classifier consisting of layers: $[input = 12 \text{ hidden} = 5 \text{ output} = 1]$, with a tan-sigmoid transfer function (Russel and Norvig, 2003) in the hidden layer and a linear transfer function in the output layer was created using Matlab (mathworks). We form cluster CB_j by extracting training examples from sensor data x_i by using time offset t from feature triplets (A,t,f) in X_j . Cluster CB_j is formed by picking 400 training examples representing $peak_1$ from sensor signal x_j at location time offset $t(3.1)$ and by picking 400 additional training examples representing $peak_2$ from sensor signal x_j at location time offset $t(5.1)$. Training examples are extracted from unprocessed sound data. Equally we form cluster CB_i by extracting training examples from unprocessed sound data x_i . As X_i contains no feature triplets, cluster CB_i is formed by picking 4000 training examples from randomly chosen time offsets from the sound signal x_i representing the sound from a healthy gearbox. We used a sliding window approach (Dias et al., 2006) when picking training examples. The window was of length 12 relating to the number of input neurons in the network and it was shifted one sample to the right each time a new sequel training example was to be obtained from a time offset. We then equally distributed cluster CB_i into sub clusters CB_{it} , CB_{iv} and CB_{ie} where t, v, e stands for training, validation and evaluation consequently. In the same manner, we make sub clusters CB_{jt} , CB_{jv} and CB_{je} from CB_j . The network was trained using supervised training and it was trained to output 1 when exposed to examples from cluster CB_j (sound data containing impact sounds) and 0 when exposed to examples from cluster CB_i (sound data not containing any impact sounds or any other abnormal sounds whatsoever).

EVALUATION

The NN classifier was trained using clusters as described in section 4 and then evaluated. In the evaluation, we focused on the ability of the NN classifier to separate between faulty and normal sound recordings. As a result of the training, the NN classifier should ideally output values closer to 1 when exposed to impact sounds and otherwise values closer to 0. However, the NN classifier are not likely to have quantized outputs. So, a simple post-processing algorithm depicted in figure 3 was applied.

For all values y in NN output:
 $y = \text{round}(y)$

Figure 3 Post-processing algorithm

We evaluated the classification performance of the NN classifier by exposing it to unprocessed sound recordings from similar gearboxes recorded during similar conditions. Sound recordings from 6 gearboxes were used for evaluation. 2 sound recordings were obtained from gearboxes containing similar faults as described in section 2 and 4 sound recordings were obtained from normal gearboxes containing no prominent impact sounds whatsoever. The NN classifier managed to achieve a correct classification score of 100%.

CONCLUSIONS

We have shown that our method successfully can be used to train back propagation neural networks on noisy sound recordings in order to classify gear faults that generates impact sounds caused by a broken gear tooth. Our approach may be usable when a simple classifier is wanted due to e.g. lack of computing power, ease of use or when only a part of the knowledge stored in the case base of a large Case-based reasoning system is needed. Further, no use of the usual sensor signal classification steps involving filtering, feature extraction and classification are needed once the neural network classifier is successfully trained. We find there is no reason not to believe that our approach would be successful for similar classification tasks.

REFERENCES

- Aamodt , A., Plaza, E., (1994), Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, *Artificial Intelligence Com.* 7, 39-59.
- Barber, A., (1992), *Handbook of Noise and Vibration Control*. 6th Edition.
- Bishop, C., M., (1995), *Neural Networks for Pattern Recognition*, Oxford : Clarendon : Oxford Univ. Press.
- Dias, F.M et al., (2006), A sliding window solution for the on-line implementation of the Levenberg-Marquardt algorithm, *Engineering Applications of Artificial Intelligence*, Pergamon.
- Lee, S., K., White, P., R., (1998), The Enhancement Of Impulse Noise And Vibration Signals For Fault Detection in Rotating and Reciprocating Machinery, *Journal of Sound and Vibration* 217, 485-505.
- Lin, J., (2001), Feature Extraction of Machine Sound Using Wavelet and Its Application in Fault Diagnosis, *NDT&E International* 34, 25-30.
- Mathworks, <http://www.mathworks.com/>
- Nilsson, M., Funk, P., Sollenborn, M., (2003), Complex Measurement in Medical Applications Using a Case-Based Approach, in: *Workshop Proceedings of the International Conference on Case-Based Reasoning*, Trondheim, Norway, pp. 63-73.
- Olsson, E., Funk, P., Xiong N., (2004), Fault Diagnosis in Industry Using Case-Based Reasoning. *Journal of Intelligent & Fuzzy Systems*, Vol. 15.
- Perner, P., (2007), Case-Based Reasoning and the Statistical Challenges, *Seventh Annual ENBIS Conference*, Dortmund, September 24-26.
- Russel, S. and Norvig, P., (2003), *Artificial Intelligence A Modern Approach Second Edidtion*, Prentice Hall International.