A framework for HVAC Malfunction Detection using Machine Learning

The Dutch Railways use air conditioning units in their trains which inherently malfunction from time to time. In this work a machine learning framework is developed to identify periods of malfunction. The K-Nearest-Neighbors, Logistic Regression and Linear SVC models were trained to identify these periods. The framework is structured such that it may be easily extended to other machine learning models and other machinery while keeping the cost of development relatively low.

Image provided by Steven Lek, Public domain, via Wikimedia Commons

Introduction

The HVAC (Heating, Ventilation and Air Conditioning) units used in NS (Dutch railways)'s trains are critical for ensuring passenger comfort: a malfunction in any one of these should be repaired as quickly as possible. Air that is too hot, too cold or stale quickly becomes agitating for any traveller.

The current method of fault detection during normal operation is fully human-driven, either via service requests by an NS employee or a complaint from a traveller. Faults may also be found during regular maintenance, which happens about four times per year. Automated malfunction detection does not yet exist for this application. Automation may aid in malfunction detection even before human discovery, which is good since malfunction detection by humans usually happens too late: Passenger comfort is already compromised in this case.

This work specifically tries to create an automated detection mechanism for cooling malfunctions in one particular HVAC type. It is then up to NS to further extend the program to their liking, be it other HVAC types or different systems altogether.

Approach

The choice for machine learning -instead of a heuristic method- was initially made because the exact definition of a malfunction was vaguely defined. The more general approach a machine learning model allows would be ideal for such a situation.

The final program was created in Python using an OOP (Object Oriented Programming) approach. This approach creates a modular program, which allows for an easy extension to other applications, and the reuse of existing modules in new applications, saving much development time. The program uses the packages scikit-learn for machine learning and Pandas for data manipulation.

The basic structure of the program can be found in Figure 2 on the following page. The runner script is the entry point for the program, and decides which datafiles are loaded and which Models are run. The Model class is used to train, store and load machine learning models, and defines the order in which input data is transformed. Each Transformer defines a single data transformation, such as handling incomplete data or applying a rolling window. These may be reused between different Model classes.

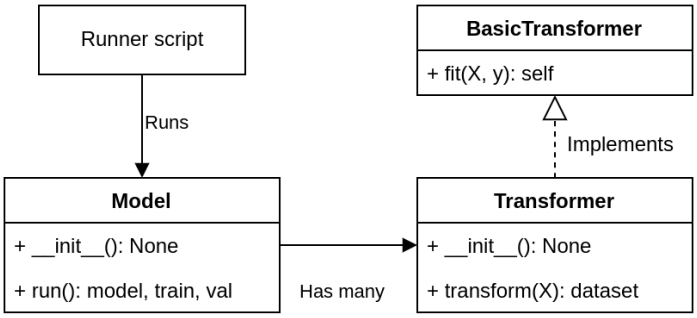

Figure 2: The generalized classes and their interaction.

The train in question consists of several coaches, each coach having an HVAC unit on either side, with thermometers both within the coach and on the outside hull. The data was presented as a time series, with one internal temperature series per HVAC, and a single overall external temperature.

Two features were extracted from the temperature data: The difference between the current temperature and the setpoint temperature and the difference between the current temperature and the median of temperatures of all coaches at that time. These are processed on a two-hour window with a one-hour step. Three different classifier models were trained on these features: A Linear SVC, a K-Nearest-Neighbors classifier and a Logistic Regression classifier. These were trained with labels given by the author and had their hyperparameters optimized for this situation. Labeling was done based on the aforementioned features.

Results

All three trained models returned good results, see Figure 3. For the cases with obvious malfunctions, a False Positive Rate of just above $1\,\%$ can be reached by all models at an acceptable False Negative Rate of between $2\,\%$ to $10\,\%$.

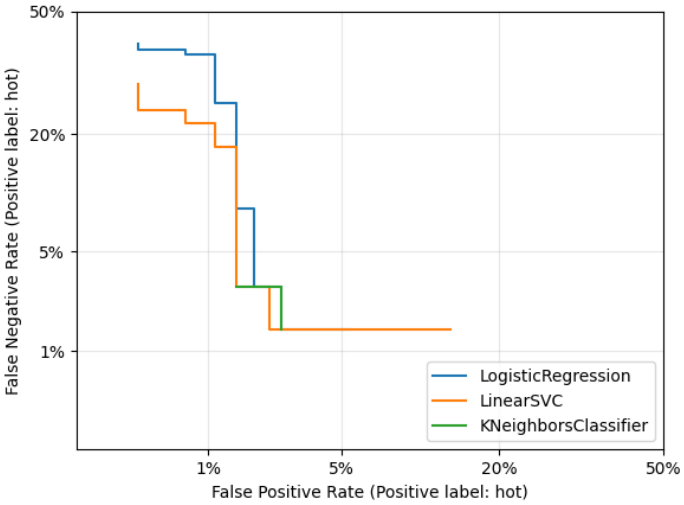

Figure 3: DET curve of the Logistic Regression (blue), K-Nearest-Neighbors (green) and Linear SVC (orange) models.

The cases where internal temperature readings are ranging from $25\,°\mathrm{C}$ to $28\,°\mathrm{C}$ were classified less well, yet below $25\,°\mathrm{C}$ the classification improves again. Based on these rates and also on the actual labeling performance, the Linear SVC model seems to work the best, not the KNN model as Figure 3 may suggest. A heuristic approach may work equally well as the machine learning model proposed in this study.

Some work is still needed to get from this malfunction detection to a fully fledged alarm system. Since the state of an HVAC is time-dependent, detected malfunctions should be validated against neighboring periods, be it the surrounding hours or days.

Conclusion

Three machine learning models were successfully trained to tackle the sketched problem of malfunction detection, of which Linear SVC is the most promising. The program created to achieve this goal is well-structured and documented, and may be easily improved or extended by NS.

UNIVERSITY OF TWENTE.

Facts

Student	Tom Veldman
University	University of Twente
Supervisors	Duncan Jansen Inka Locht Mariëlle Stoelinga Mark Vlutters
Company	Dutch Railways

PrimaVera - Powered by:

NWO

Full text:
purl.utwente.nl/essays/89342