Antti Mäkelä

Tuomo Sipola

Detecting gear differences based on KPIs

Report for eÄlytelli project in Information Technology

November 26, 2021

JAMK University of Applied Sciences

Information Technology

Preface

eÄlytelli project develops knowledge and information about the possibilities of IoT's data analysis for the use of companies and other actors by building a research and testing system for the needs of research and education. The operational environment is developed so that it can be used as a learning platform suitable for the needs of degree programmes. With the project, awareness is raised about the opportunities of IoT among diverse actors and at the same time, the know-how of the project executors is increased, which improves the opportunities to train new experts. The purpose of the project is to develop and pilot concrete data analysis solutions from selected fields of business in order to create new business, achieve growth in the already existing business as well as to increase the number of jobs. The developed testing environment gives good possibilities to do differents of kind of AI pilots, integrates users to the trials, supports setup phase and needed changes, shows the advantage of the solutions to the potential customers, and gives a solid foundation for the distribution of the developed methods.

List of Figures

Figure 1. Example of KPCA locations of the machines 4

Contents

1 WP3 OBJECTIVES 1
1.1 Objectives 1
1.2 Fullfilling the objectives 1

2 WP3 RESULTS 2
2.1 Identified needs 2
2.2 Data sources 2
2.3 Analytics 2

3 CONCLUSION 5

BIBLIOGRAPHY 6

1 WP3 Objectives

Work package 3 *Advanced data analysis* of the eÄlytelli project had the following objectives, as set in the Project Plan dated 10 March 2020. In the Cooperation Agreement dated 29 January 2020, work package 3 duration is set from Q1/2021 to Q4/2021.

1.1 Objectives

- Task 3A: Reviewing needs. Deliverable: Specification.
- Task 3B: Setting goals. Deliverable: Specification.
- Task 3C: Data source readiness. Deliverable: Interface implementation.
- Task 3D: Analytics. Deliverable: Code implementing analytics methods.
- Task 3E: Visualization and reporting. Deliverable: Code implementing visualization (no UI).
- Task 3F: Publication writing and publishing. Deliverable: Publishable report.

1.2 Fullfilling the objectives

This document specifies the goals identified in Task 3A and Task 3B. This work was done during the project, and the needs were identified during recurring meetings.

Data source connections have been implemented for Task 3C in the codebase.[1] This includes data reading from TDMS files and from the database, as well as writing results to the database. In addition, an intermediate data dump is possible for analytics, and this feature was used in Tasks 3D and 3E.

The following chapter 2 details the analytics used for Task 3D and the visualization results of Task 3E. The code implementations are available in WP3 data analysis repository.[2]

This report serves as the delivarable of Task 3F.

1. https://gitlab.labranet.jamk.fi/ealytelli/moventas/
2. https://gitlab.labranet.jamk.fi/ealytelli/wp3dataanalysis/

2 WP3 results

2.1 Identified needs

The identified use case was related to the identification of problematic ramp measurements. The value in identifying them is in that a faulty unit or faulty measurement can be detected much sooner and faster.

2.2 Data sources

Data sources were discussed in the report of WP1 Raitapuro and Sipola 2020, which included only saving the target data for processing. During WP3, the ability to save results to a database was added. The data saving is configurable so that the database address can be changed if needed.

Data processing code to produce phenomena reports and respective phenomena curve data is available in JAMK's GitLab in `moventas` repository.[1] This data processing is detailed in the report of the previous WP2 Sipola 2021. The data processing program was maintained and updated during WP3. Among the added feature are:

- Reading data from TDMS file in a configurable way.
- Reading data from SQL database.
- Saving results to SQL database.
- PDF report generation.

2.3 Analytics

The phenomenon amplitude time series extracted from the input data were used to perform more sophisticated analysis. The primary tool utilised during the data analysis process was *Jupyter Notebook* that provided a fast feedback environment for *Python* based analytics. Several Python data analysis libraries were used for wider range of analysis methods available.

1. https://gitlab.labranet.jamk.fi/ealytelli/moventas/

Pandas data processing library was used to efficiently manage and manipulate large quantities of tabular data. *scipy* scientific computation library provided several useful mathematical algorithms used in the analysis. *matplotlib* library provided the means to visualise the data with several different types of visualisations, such as scatter plots or line plots. *scikit-learn*, a machine learning library for Python, contains several machine learning algorithms for regression, clustering and dimensional reduction.

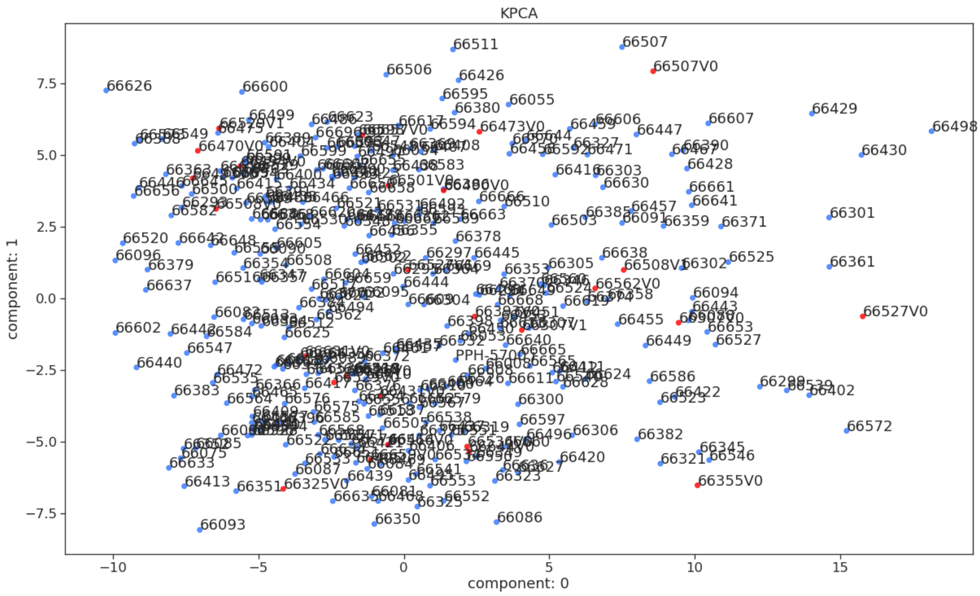
The input data contained hundreds of phenomenon amplitude time series for each data point. The total amount of scalar values per data point exceeded tens of thousands. Preprocessing was performed in order to reduce the high dimensional data to more manageable: the time series were segmented and each of the segments were reduced to values describing the distribution of the segment, such as mean, median, standard deviation and skew. These statistical values formed new data points that were passed to *principal component analysis* (PCA) dimensional reduction algorithm. Both linear PCA and kernel PCA with different parameters were used to provide low dimensional views into the data. The dimensionally reduced data was visualized with scatter plots.

Results of the PCA was further analysed with a significant feature extraction process. Each data point had a binary label that could be used to split the dataset into two groups. By dropping certain features from the input data, one may observe whether the resulting analysis has weaker or stronger correlation with the labels. *Mahalanobis distance* was used to determine the overall distance between the two groups: the smaller the distance was, the more significant that particular dropped feature was in grouping the data points. By iteratively going through all input features in this manner allows the features to be ranked in order of significance.

Neural networks, such as autoencoder networks, were superficially trialed as a part of the analytics process. However, the results from the initial experiments were inadequate due to neural network over fitting and the research focus was shifted to principal component analysis.

Figure 1 shows an example of the embedded space. Each data point represents one machine. As can be seen, no clear distinction between nominal and anomalous machines can be found.

Figure 1. Example of KPCA locations of the machines. Blue dots represent normal machines, while red dots represent known problematic cases.



3 Conclusion

WP3 achieved its goals. Data sources were handled as needed, and the pipeline was streamlined with new features. The conducted research showed that autoencoders and PCA are possible candidates for classification of problematic ramp measurements. Currently, the features seem to be insufficient to discriminate between machine classes. Further development could try different features before using PCA.

Bibliography

Raitapuro, Jesse, and Tuomo Sipola. 2020. *Dedicated data storage suitable for efficient visualization and analysis: Report for eÄlytelli project in Information Technology.* Technical report. Jyväskylä: JAMK University of Applied Sciences.

Sipola, Tuomo. 2021. *Frequency phenomena analysis and visualization: Report for eÄlytelli project in Information Technology.* Technical report. Jyväskylä: JAMK University of Applied Sciences.