Linköping University | Department of Computer and Information Science
Master thesis, 30 ECTS | Computer Science
202018 | LIU-IDA/LITH-EX-A--2018/XXX--SE

BEAUTIFUL TITLE, ENGLISH

– SUBTITLE, ENGLISH

VÄLDIGT FIN TITEL

Linus Kortesalmi

Supervisor : Mattias Tiger
Examiner : Fredrik Heintz

External supervisor : Simon Johansson

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

Upphovsrätt

Detta dokument hålls tillgängligt på Internet – eller dess framtida ersättare – under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår. Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns lösningar av teknisk och administrativ art. Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart. För ytterligare information om Linköping University Electronic Press se förlagets hemsida http://www.ep.liu.se/.

Copyright

The publishers will keep this document online on the Internet – or its possible replacement – for a period of 25 years starting from the date of publication barring exceptional circumstances. The online availability of the document implies permanent permission for anyone to read, to download, or to print out single copies for his/hers own use and to use it unchanged for non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional upon the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility. According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement. For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: http://www.ep.liu.se/.

© Linus Kortesalmi

Abstract

Abstract.tex

Acknowledgments

Acknowledgments.tex

Contents

Abstract **iii**

Acknowledgments **iv**

Contents **v**

1 Introduction **1**
1.1 Problem Description 1
1.2 Aim 1
1.3 Research Questions 2
1.4 Delimitations 2

2 Theory **3**

3 Related Work **4**
3.1 Multiple Classifier Approach 4
3.2 title 4

4 Method **5**

Todo list

The RQs below are just an initial draft. 2
Results of the Multiple Classifier approach? 4
Requirements in order to use it? (type of data, dimensionality, environment (R2F), supervised/unsupervised?, etc. 4

1 Introduction

Maintenance of devices/vehicles/systems/machines (hereinafter referred to as "units") is generally done by planned scheduling. Typically this is done when some parameter of the system reaches a threshold value. For example, car maintenance can be scheduled after 30,000 kilometres, after a year or perhaps after a certain number of operating hours. One problem with planned scheduling is the reliance on experience and statistics from many units. For a single unit, the planned scheduling will either be executed too early (could have waited longer before service) or too late (problems encountered before threshold reached).

Internet of Things (IoT) permits the streaming of continuous data from multiple units. The data is typically the state of the unit in the shape of many different variable values. A rule framework can be built incorporating the continuous data. The framework can then give an informed service alert based on the actual state and need of a unit.

Machine Learning (ML) models can be used to, for example, capture dependencies in large-scale data sets (ref needed), anomaly detection (ref), clustering (ref), image recognition (ref), and decision making (ref). Anomaly detection could be used together with continuous real-time data from a system to find unusual changes or behaviour, which could be the basis for a service alert and/or the gathering of new knowledge about a system. ML-algorithms can also be used to perform Predictive Maintenance (PdM).

1.1 Problem Description

1.2 Aim

The expected results is a study on how ML-algorithms can be used for Predictive Maintenance (PdM). The study shall compare at least two different ML-algorithms. The goal is to implement the ML-algorithms and use real-world data from an existing IoT system. The aim of the study is also to compare how well the improved DcM performs with the current solution used in the IoT system (preliminary, more info needed from company as well as ideas how to compare the solution with the practice in use today).

1.3 Research Questions

This work explicitly answers the following questions:

The RQs below are just an initial draft.

1. Which methods can be used to detect anomalies in a dynamic real-time system?
2. How can the need for maintenance be detected from real-time processing of (domain-specific) data?
3. How can predictive maintenance be achieved by processing real-time (domain-specific) data?
4. How far in advance can maintenance be predicted?

1.4 Delimitations

2 Theory

3 Related Work

Predictive Maintenance (PdM) approaches have been studied and tested for many different industries. This chapter covers some of the approaches proposed in the literature.

3.1 Multiple Classifier Approach

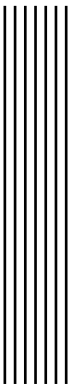
In [1], Susto et al. propose a multiple classifier approach to PdM. They use the approach to predict problems which stem from the "wear and tear" effects of equipment used for semiconductor manufacturing. Each classifier is trained on a different failure horizon m, which results in a different classification problem for each classifier. The failure horizon is the number of iterations in a maintenance cycle where the fault has taken place. In a traditional R2F (Run to Fail) environment only the last iteration would be faulty ($m = 1$). Instead, the dataset is transformed for each classifier so that the last m iterations are marked as faulty. A larger m reduces the skewness of the dataset and enables a more conservative PdM policy. The multiple classifier approach thus enables the implementation of a cost optimisation policy as well as a fault prevention policy.

Results of the Multiple Classifier approach?

Requirements in order to use it? (type of data, dimensionality, environment (R2F), supervised/unsupervised etc.

3.2 title

4 Method



Bibliography

[1] G. A. Susto, A. Schirru, S. Pampuri, S. McLoone, and A. Beghi. "Machine Learning for Predictive Maintenance: A Multiple Classifier Approach". In: *IEEE Transactions on Industrial Informatics* 11.3 (June 2015), pp. 812–820. ISSN: 1551-3203. DOI: 10.1109/TII.2014.2349359.