

54th CIRP Conference on Manufacturing Systems

Predictive analytics in quality assurance for assembly processes: lessons learned from a case study at an industry 4.0 demonstration cell

Peter Burggräf^a, Johannes Wagner^a, Benjamin Koke^a, Fabian Steinberg^a, Alejandro R. Pérez M.^a, Lennart Schmallenbach^a, Jochen Garcke^{b,c}, Daniela Steffes-Lai^b, Moritz Wolter^{b,d}

^aChair of International Production Engineering and Management (IPEM), Universität Siegen, Paul-Bonatz-Straße 9-11, Siegen - 57076, Germany

^bFraunhofer Institute for Algorithms and Scientific Computing (SCAI), Schloss Birlinghoven 1, Sankt Augustin- 53757, Germany

^cInstitut for Numerical Simulation, Universität Bonn, Endenicher Allee 19b, 53115 Bonn

^dInstitut for Computer Science, Universität Bonn, Endenicher Allee 19a, 53115 Bonn

* Corresponding author. Tel.: +49-271-740-4509; fax: +49-271-740-2630. E-mail address: alejandroperez@uni-siegen.de

Abstract

Quality assurance (QA) is an important task in manufacturing to assess whether products meet their specifications. However, QA might be expensive, time-consuming, incomplete, or delayed. This paper presents a solution for predictive analytics in QA based on machine sensor values during production while employing machine-learning models based on logistic regression in a controlled environment. Furthermore, we present lessons learned while implementing this model, which helps to reduce complexity in further industrial applications. The paper's outcome proves that the developed model was able to predict product quality, as well as to identify the correlation between machine-status and faulty product occurrence.

© 2021 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Peer-review under responsibility of the scientific committee of the 54th CIRP Conference on Manufacturing System.

Keywords: Machine-Learning; Predictive Quality; Production; Quality Assurance; Logistic Regression

1. Introduction

(AP:

- Products' quality must always be guaranteed.
- In certain cases (especially in series assembly production) there is a delay between quality inspection and end of the production process high potential for producing scrap
- By doing quality inspection based on samples, there is still a risk of sending NOK parts to customers
- The cost of quality checks might be expensive
- Quality assurance is normally done at the end of production (for assembly processes)
- Early prediction of the production's quality would help to decrease the amount of scrap and the risk of sending NOK parts to customers
- Variable predictions by using machine learning models is currently widely used in diverse fields due to its

accurate predictions with the data collected from the machines, prediction models can be implemented for specific predicted values

2. Related Works

2.1. Quality check in production

(AP:

- How quality check is done in production
- Concepts and sampling plan
- Attributive vs Variable measurements [VM has higher measurement accuracy, reason why it's better to use regression model]

Fig. 1. The industry simulation cell. (MW: TODO: Find image.)

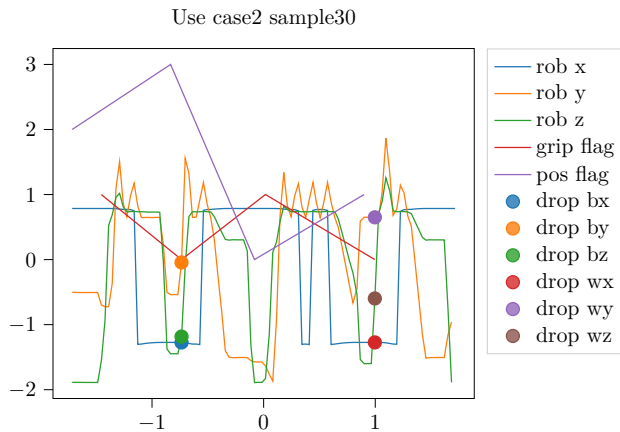


Fig. 2. Robot arm movement patterns. We show the movement of the cell's robot arm, its grippers and an encoding of the overall position in the system. The changes in the grip flag are triggered, when the gripper opens or closes. Zero indicates an open, while one indicates a closed gripper. The position flags marks the state of the cell's arm. 0 means the black disk is in store, one that the black disk is in the gate, two that the white disk is in store and three that the white disk is located in gate.

2.2. Predictive analytics in production

(AP: Predictive analytics as concepts. Implementation in production)

- Predictive Maintenance
- Scheduling
- Other

If happens to be something with P.A. Quality Check

- Predictive analytics in production focusing in QA
- Existing approaches and methods

Note: Ideal is to find a gap in the literature, that proves why do we do it and why it is important)

3. The case industry 4.0 demonstration cell

(AP: What is the demozelle and why it is relevant as study case Description of the assemble process in the Demozelle Incl pictures)

4. Methods

Data is collected from the demonstration cell and analyzed

Support Vector Machine	95.%
Multilinear Perceptron	95.%
Random Forest	100.%

Table 1. Method comparison for our anomaly detection task using the industry demo data test set.

4.1. Measurement

4.2. Machine Learning

We compare Support Vector Machines (SVM), Multilinear Perceptrons [1] as well as Random forests in the task of quality prediction.

5. Experiments

5.1. Recording the data

5.2. Classifier optimization and testing

The robot's position at disc drop is marked in Figure 2. The quality prediction problem is framed as a classification task. The input vectors which we feed into our classifiers consists of the arm positions at the disc drops for both discs in three dimensions, as well as the largest recorded belt speed of all three belts. The interpretation of the quality measurement is used as the training target. It can be good or bad which we encode as zero and one.

We work with a total of 132 measurements. Each containing arm and belt data logs of individual cell runs. 20 samples are set aside at random for testing purposes, leaving 112 training samples. The random number generator seed is set to one in order to ensure the train and test set splits are identical for all experiments.

We compare a total of three different classifier architectures on the data. A Support Vector Machine (SVM), a multilinear perceptron and a Random Forest structure. All three methods are trained using standard hyperparameters.¹ Results are shown in table 1. For 55 of the total 132 samples quality measurements indicate a problem. This sets the baseline over the entire data set, that would be obtained by simply labeling all samples as good. Over the entire data set, we require to classify more than 58.3% of the data correctly. The test set contains eight incorrect samples. We would therefore expect any naive classifier to produce a 60% accuracy, which we have to beat. In Table 1, we observe that this is indeed the case for all three approaches evaluated here. The random forest performed best in this case.

6. Conclusion

6.1. Lesson Learned

(AP: explain experiences by doing the research)

¹ In order to allow exact reproduction of these results source code is available at <https://github.com/manubrain/Demo-Cell-Classification>

7. Conclusion

References

- [1] C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.