Supplementary quality control features for the production department in Odoo ERP

A S Ahmadiyah1,2, Y Y Ratna1,3, N N Yotifa1,4, and I Dinillah1,5

1Informatics Department, Sepuluh Nopember Institute of Technology, Surabaya, Indonesia

E-mail: 2adhatu@if.its.ac.id, 3yukiratna.17051@mhs.its.ac.id, 4nitama.17051@mhs.its.ac.id, 5izzahdinillah.17051@mhs.ac.id

Abstract. Odoo is one of the top Enterprise Resource Planning (ERP) applications. In particular, it is supported by a quality control module with the ability to control, trigger an alert, and check its purpose. This research strives to supplement the existing Odoo quality control module with parameters from past inspection data obtained before quality control migrates to Odoo. To support the study, we used a machine learning method to discover the intrinsic pattern from the dataset of quality control in the production process of baby biscuit products. The experiment shows that the additional quality control feature can provide the product measurement and tolerance threshold fed into Odoo quality control module. The additional feature is helpful for decision making and error minimization in setting quality control parameters and tolerance threshold. Furthermore, a high accuracy rate of 95.71% is obtained from the employed Decision Tree algorithm.

1. Introduction
Quality control (QC) is vital to ensure that the final product or service meets the requirements and a set of quality standards. It supports companies to maintain and protect the quality of products or services, which leads to achieving the target market opportunity through customer satisfaction [1]. Specifically, in the manufacturing industry, quality control appears as quality control of raw materials, quality control in the production process, and quality control of the final product.

In a large-scale business, quality control is recorded into an integrated management system such as an Enterprise Resource Planning (ERP). Odoo is one of the popular open-source ERP tools. Odoo is equipped with quality inspection in which one can create an inspection plan based on business processes and company needs. The role of ERP in quality inspection is to speed up and simplify the inspection process.

Odoo quality control consists of three functions, i.e., control, check, and alert functions. In defining the quality control point using take measure type, one needs to input product measurement along with the threshold value during the production process. This feature can only be used for the currently inspected data. However, this feature does not accommodate past data before the company migrates quality control to Odoo, which potentially holds essential information about measurement and threshold.

This paper aims to tackle this problem by suggesting additional quality control features by extracting inherent information from the past data. We utilized a machine learning method to enable the learning process. Machine learning is a powerful tool to conduct quality inspection [2][3][4][5][6][7]. Our proposed solution can automatically find the product measurement and threshold values fed to the Odoo quality control module by using the baby biscuit production process in our experiment.

2. Methodology
Our concern is to incorporate essential information from past inspection data generated before the QC migration to Odoo as a recommendation fed to define quality control points. As seen in Figure 1, our proposed process is illustrated in the bottom left side. It starts with the preparation of the past inspection dataset. Then, a Decision Tree algorithm is utilized to discover intrinsic patterns. Finally, the
recommended product measurement and threshold from the past is obtained. The term past refers to quality control conducted without the Odoo quality control module.

**Figure 1.** The position of our proposed process against the existing Odoo quality control module.

As seen in Table 1, the past inspection dataset from baby biscuit is prepared. From 233 collected records, the quality control dataset has several dimensional measurements or attributes. They are shape, diameter in centimeters, weight in grams, and color in RGB, separated into three-color channels ranging from zero to 255. The products are divided into five codes; there are M01, M02, M03, M04, and M05; each represents the flavor of the baby biscuits.

**Table 1.** Samples of baby biscuit quality control in production process.

| Product Code | Circular Shape | Diameter (cm) | Weight (gm) | Color Channel |   |
|--------------|----------------|--------------|-------------|---------------|---|
| M03          | Yes            | 5.9          | 9           | 191           | 173 | 109 |
| M02          | Yes            | 5.8          | 8.9         | 189           | 179 | 98  |
| M04          | Yes            | 6            | 10.83       | 192           | 161 | 101 |
| M01          | Yes            | 4.9          | 10.83       | 193           | 173 | 111 |
| M05          | No             | 5.6          | 11          | 213           | 182 | 101 |
| M05          | Yes            | 5.7          | 12.1        | 190           | 162 | 101 |
| M04          | Circle         | 6            | 8           | 189           | 175 | 110 |
| M04          | Circle         | 5.8          | 12.5        | 201           | 170 | 104 |

Next, the discovering intrinsic patterns step is performed using the Iterative Dichotomiser 3 (ID3) algorithm [8], one type of the Decision Tree, to train each data in the dataset. The ID3 algorithm generates a tree structure. At first, The ID3 selects the root node, which has the highest information gain value. It is followed by setting the next attribute with the highest information gain value among the rest of the attributes as the branch node. This process is repeated until all attributes are counted.
From this step, we have a model to classify each data into pass or defect type. This model's performance is then tested to ensure the accuracy of the model, leading to trusted results fed into the Odoo quality control module. Lastly, we calculate product measurement tolerance from the model built, which is also needed by the Odoo quality control module.

3. Results and Discussions
From 233 records used in the training phase, the classifier can predict 95.71% of the quality control status correctly, leaving the 4.29% misclassified. Meanwhile, Table 2 shows the result of the testing dataset using the model built using the ID3 algorithm. The column labeled Expected Output is the result of the past inspection. In contrast, the column labeled Real Output represents the result obtained using ID3. In this case, The ID3 algorithm is proved to perform well in revealing the intricate pattern of the dataset.

Table 2. Quality control test results.

| Test # | Product Code | Circular Shape | Diameters (cm) | Weight (gm) | Color Channel | Expected Output | Real Output |
|--------|--------------|----------------|----------------|-------------|---------------|----------------|-------------|
| 1      | M03          | Yes            | 5.6            | 10          | Red 203       | 177            | 101         | Pass        | Pass        |
| 2      | M02          | Yes            | 5.8            | 9           | Green 210     | 172            | 101         | Pass        | Pass        |
| 3      | M04          | Yes            | 5.7            | 11          | Blue 205      | 181            | 109         | Pass        | Pass        |
| 4      | M01          | Yes            | 4.9            | 10          | Red 201       | 170            | 109         | Defect      | Defect      |
| 5      | M05          | No             | 5.8            | 9           | Red 202       | 180            | 109         | Defect      | Defect      |
| 6      | M05          | Yes            | 5.6            | 10          | Red 216       | 170            | 105         | Defect      | Defect      |
| 7      | M04          | Yes            | 5.7            | 8           | Red 201       | 171            | 102         | Defect      | Defect      |
| 8      | M01          | No             | 5.8            | 9           | Blue 191      | 179            | 109         | Defect      | Defect      |

In Table 3, we organized product measurement and tolerance threshold similar to the Odoo quality control module, particularly defining the quality control point feature. The product measurement name, type, and unit of measurement (UoM) are obtained directly from the dataset attributes. In contrast, the minimum, maximum, and tolerance value of the quantitative type are obtained from machine learning, in our case, the ID3 algorithm. Holding that information, the quality control team has the option to directly use the recommendation from our supplementary quality control feature or make adjustments. This can lead to minimizing the effort of setting quality control points and human error.

Table 3. Product measurement and tolerance threshold.

| Name           | Type       | Minimum tolerance | Minimum | Maximum | Maximum tolerance | UoM | Qualitative value |
|----------------|------------|--------------------|---------|---------|-------------------|-----|-------------------|
| Circular shape | qualitative| 0                  | 0       | 0       | 0                 | -   | 2 records (Yes, No) |
| Diameters      | quantitative| 0.2               | 5.8     | 5.8     | 0.2               | -   | centimeters       |
| Weight         | quantitative| 2                 | 10.83   | 10.83   | 0.2               | 2   | grams             |
| Red channel    | quantitative| 0                 | 185     | 220     | 0                 | -   | -                 |
| Green channel  | quantitative| 0                 | 140     | 165     | 0                 | -   | -                 |
| Blue channel   | quantitative| 0                 | 80      | 80      | 0                 | -   | -                 |

To be used by other companies, this solution can be adopted directly. However, the high accuracy of machine learning may vary due to the inspection of the dataset's characteristics. The accuracy of the
model built under different inspection dataset needs to be checked before continuing to be filled into the Odoo quality control module.

4. Conclusion
When the quality control is established long before migrating to Odoo, the past quality control data can be extracted automatically using a machine learning method, specifically the ID3 algorithm. Usually, the accuracy rate is the one sought. Furthermore, the splitting criteria from the ID3 algorithm are beneficial in providing the tolerance threshold.

References
[1] Tata J, Motwani J and Prasad S. Benchmarking Quality Management Practices: US Versus Costa Rica. *Multinatl. Bus. Rev.* 2000;8(2):37-42. Retrieved from: https://www.researchgate.net/publication/292017239_Benchmarking_quality_management_practices_US_versus_Costa_Rica
[2] Anand S, Priya L. *A Guide for Machine Vision in Quality Control.* Chapman and Hall/CRC; 2019 Dec 23; doi.org/10.1201/9781003002826
[3] Mohammadi P, Wang ZJ. Machine learning for quality prediction in abrasion-resistant material manufacturing process. *2016 IEEE Can Conf Electr. Comput. Eng. (CCECE).* 2016 May; doi.org/10.1109/ccece.2016.7726783
[4] Ordukaya E, Karlik B. Quality Control of Olive Oils Using Machine Learning and Electronic Nose. *J. Food Qual.* 2017;2017:1–7. doi.org/10.1155/2017/9272404
[5] Alonzo LMB, Chioson FB, Co HS, Bugtai NT, Baldovino RG. A Machine Learning Approach for Coconut Sugar Quality Assessment and Prediction. *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Inf. Technol, Commun Control. Environ. Manag (HNICEM).* 2018 Nov; doi.org/10.1109/hnicem.2018.8666315
[6] Li F, Wu J, Dong F, Lin J, Sun G, Chen H, et al. Ensemble Machine Learning Systems for the Estimation of Steel Quality Control. *2018 IEEE Int Conf on Big Data* (Big Data). 2018 Dec; doi.org/10.1109/bigdata.2018.8622583
[7] Bordekar L, Velingkar H, Fernandes E, Bandekar HH, Harmalkar AG, Antonio Pinto BJ. Cashew Nut Grade Identification and Quality Testing Using Machine Learning. *2018 Second Int. Conf. Inventive Commun. Comput. Technol (ICICCT).* 2018 Apr; doi.org/10.1109/icicct.2018.8473058
[8] Quinlan, J.R. Induction of decision trees. *Mach Learn 1.* 1986:81–106. doi.org/10.1007/BF00116251. Retrieved from: https://link.springer.com/article/10.1007/BF00116251