engineering proceedings

MDPI

Proceeding Paper

Machine-Learning-Based Approach to Orange Quality Detection and Classification †

Aiman Hameed [1,*], Tahir Muhammad Ali [2] and Hermanto [3]

[1] Department of Software Engineering, University of Sialkot, Sialkot 51040, Pakistan
[2] Department of Computer Science, Gulf University for Sciences and Technology, Safat 13060, Kuwait; ali.t@gust.edu.kw
[3] Informatics Engineering, Nusa Putra University, Sukabumi 43152, West Java, Indonesia; hermanto@nusaputra.ac.id
* Correspondence: 21101001-110@uskt.edu.pk
† Presented at the 7th International Global Conference Series on ICT Integration in Technical Education & Smart Society, Aizuwakamatsu City, Japan, 20–26 January 2025.

Abstract

Oranges are the most used fruits in Pakistan. Ensuring quality in oranges will definitely enhance customer satisfaction, and it will do wonders with respect to marketing. In the old days, this consisted of physical inspection with one's hands and laboratory tests that were very slow and inaccurate in various cases and whose quality was not admissible for any large-scale application. The projected straightforward and simple method identifies pertinent features such as acidity, vitamin C, texture, and color. We used random forest, decision tree, KNN, and naive Bayes, with which we acquired 91.1%, 93.4%, 92.32%, and 97.67% of the data correctly, where naive Bayes performed the best. These methods induce speed, accuracy, and scalability for quality assurance in oranges and are feasible for use in citrus farming. The future will be focused on developing low-cost approaches for smallholder growers and dealers working towards revolutionizing orange quality testing.

Keywords: orange quality; fruit quality assessment; machine learning; naive bayes; citrus farming

check for updates

Academic Editors: Debopriyo Roy, George F. Fragulis and Peter Ilic

Published: 15 September 2025

Citation: Hameed, A.; Ali, T.M.; Hermanto. Machine-Learning-Based Approach to Orange Quality Detection and Classification. *Eng. Proc.* **2025**, *107*, 92. https://doi.org/10.3390/engproc2025107092

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Oranges are some of the most famous fruits of all fruits because of their taste. Quality in oranges is very important to make the customer happy, to meet market standards, and reduce losses during storage and transport [1]. Orange quality depends on features of their appearance like their size, color, and texture also internal characteristics like sweetness, acidity, and vitamin C. The traditional methods for checking orange quality rely on manual testing. Here, we also studied previous papers on this topic whose data is limited and not accurate. We used a voting system to increase the accuracy, in which we used four different methods, which are named random forest, decision tree, KNN (K Nearest Neighbor), and naive Bayes. To extract features for better quality, we used brute force. After voting, we acquired a higher accuracy of 97.67% with naïve Bayes; also, for the other models applied, we achieved 91.1% for random forest, 93.4% for decision tree, and 92.32% for the KNN model.

2. The Literature Review

The taste of different kinds of oranges has been studied in Vietnam's Mekong Delta and Kon Tum. These studies have found the best kinds for eating and using in things, with big shifts—Cam Canh is very sweet, and Cam Sanh is high in ascorbic acid. This gave a new look at how oranges change in these locations and paths to make them better [2]. Shravan and his team are working on looking at the changes in sweet oranges from Maharashtra, India, as they ripen. They have found significant changes, like a greater sugar content and less acid in the fruit. This research points out how good these fruits are for the body and their benefits for farming and making food [3]. Çetinkaya and his team looked at how to keep Jaffa oranges fresh over a long-term duration in cold storage. They dealt with the problem of the Mediterranean fruit fly. Their work showed that a small dose of radiation kept the fruit in a good state and preserved the taste. This offered a new perspective on how to make citrus foods [4]. Apte and Patavardhan undertook the difficult task of testing good fruit. They looked at the outside color, while the inside was compared using X-ray marks. This improved upon old methods that only detected the skin. Their work is indicative of a research gap in previous works which just checked the shell and omitted the interior. They mixed color shots, Gabor filters, and HSV points. They combined these with Support vector Machine (SVM) and Artificial Neural Network (ANN) types of classifiers to classify oranges and bananas into three good classes. Here, this time, SVM was 100% right. This improved upon earlier work by Komal in 2019 which only looked at the outer layer. This proves that for proper fruit packing, multi-method usage is best [5]. Cetinkaya and his team members are working on using gamma rays to protect Jaffa oranges from the Mediterranean fruit fly, a big problem when growing citrus. They saw that doses of up to 1.0 kGy kept the fruit in good shape, and 0.5 kGy was the top pick. This method helps the fruit stay fresh and look nice [6]. Another research paper has looked at how well light doses (of 1.5 kGy) of radiation work on Jaffa oranges, aiming to keep them good-quality in cold storage. The team finds that light doses of 1.0 kGy preserve the fruit's quality and do not change their market value. This matches what past tests on other citrus fruits around the world have shown [1].

Quaggio and his team members looked at how sweet oranges grow and how the taste of orange grown in hot places is affected when substances like nitrogen, phosphorus, and potassium are added to the soil. They saw that the best mix made the oranges bigger and tastier than the others. Yet too much potassium can hurt the fruit [7]. Zeb's group has worked on using short-wave Near-Infrared Spectroscopy (NIR) technology to assess the sugar content in three varieties of oranges from Pakistan, blood red, Mosambi, and Succari, thus addressing an unexplored area in previous works. They have worked on using AI to classify the sugar levels with a direct sorting accuracy of 81.03% and derived credible values for mixed varieties too [8]. Seminara and her team are all working on sweet orange cultivation challenges like Huanglongbing disease and low genetic diversity, emphasizing the need for adaptable fruit and modern breeding techniques [9]. Bhusal and his team researched the sales and growth of sweet oranges in Nepal and suggested that poor post-harvest care and a lack of water and cheap crops available to other farmers, as well as imports from other countries, contribute to market competition. They recommend that extra water, fresh seeds, and proper storage could increase revenue [10].

3. Methodology

Machine learning is a sub-branch of artificial intelligence that uses multiple algorithms to train and test models utilizing previous data [11] as the input and applying different algorithms, for example, here, algorithms such as KNN, naive Bayes, random forest, and

decision tree are applied to predicting the best model results—primarily used in software applications [12,13].

3.1. KNN

KNN stands for K-Nearest Neighbor. It is a supervised machine learning (ML) algorithm that is used to classify a relevant group based on its nearest group. The K value is used to achieve good accuracy in the model. It works based on the nearest class with the most data from the same class and knows that the k value must be odd. It is an example of a lazy learner algorithm [14].

3.2. Decision Tree

Decision tree (DT) is a supervised ML algorithm that is most commonly used in decision-making and data analyses to recognize the way to achieve the aim for a given situation [15]. It is a tree-based model and is used for both classification and regression. A tree's internal nodes represent the test attributes, branches represent the outcomes, and leaf nodes represent the class labels in the model.

3.3. Naïve Baise

The naive Bayes algorithm is a crucial algorithm that helps with classification problems. It is derived from Bayes' probability theory and is used for text classification, where training proceeds on high-dimensional datasets and acquires good accuracy.

3.4. Random Forest

The random forest algorithm in machine learning is an ensemble learning method that combines multiple decision trees, and each is trained on a random subset of data and features to produce a more accurate prediction by taking the majority vote from all of the trees, making it effective for both classification and regression tasks [16].

3.5. The Dataset

The dataset we used to test machine learning had many parts to check and estimate orange quality. These parts were size in cm, weight in grams, brix for assessing sweetness, and pH for assessing sourness. Other features added were softness and ripeness, with both scored from 1 to 5, and days since picking as the harvest time. The dataset included color and type to describe appearance and type; an indicator to say whether an orange had spots or not, given as yes or no; and quality as an all-around score from 1 to 5. These parts constituted a full base for learning and testing out the machine learning methods [17].

3.6. The Data Split

The Split Data operator in RapidMiner is used to split a provided set of data into a specified number of subsets. This proves to be useful while splitting data for machine learning usage, such as model training and testing. The split ratios may be set by the users to whatever they wish, but they need to ensure that the sum of all ratios is equal to 1. A common split, for example, is to use 70% of the data to train with and use 30% to test with. This gives the model enough to learn from the training set but enough for testing with unseen data in the test set.

Compared to some of the other RapidMiner operators, the Split Data operator has the capability of randomized and adaptive splitting, ensuring that all subsets are a true representation of the original data distribution. The Split Data operator also supports deterministic and non-deterministic splits depending on whether a random seed is used or not. The Split Data operator is a significant player in the data preprocessing process and helps improve the reliability and performance measurements of machine learning models.

3.7. Brute Force

Brute force was used for feature selection. A brute force algorithm solves a problem through an exhaustive search—it tries every possibility until it finds the optimal subset. Computationally expensive though this is, it ensures that no potentially perfect feature subset is left out. The model was initially trained on the complete feature set, and then brute force was used to examine the contribution of each feature to the performance of the model.

This approach facilitated the recognition of the most relevant and useful features, and it significantly improved the efficiency and accuracy of the model. Brute force feature selection is particularly useful when the feature dimensionality is relatively low since a globally optimal solution is certain. It is simple to implement and has a tendency to serve as a baseline when evaluating more complex or heuristic-based feature selection techniques. Although brute force is computationally costly, it is a certain method for gaining knowledge on which how important a part features contribute to a dataset.

4. Results

This sub-section offers a comprehensive overview of the results obtained upon applying the various machine learning and deep learning models. The performance of every model was evaluated using a Performance Vector, which presents an overall summary of important evaluation metrics. These include accuracy, precision, recall, the F-measure, and others such as the AUC (Area Under the Curve) when appropriate.

In addition to this, a confusion matrix for each of the models was derived to present the correct and incorrect predictions on the test set. The matrix gives an idea of the accuracy with which the model classifies different classes. On observing these values for the performance and matrix outcomes, we can figure out the strengths and weaknesses of each model and pick out the model that provides us with the most accurate and reliable predictions for our specific problem.

4.1. Accuracy

Accuracy is the most widely used metric to verify the general performance of a machine learning model. It informs us about how often the model is correct in its predictions out of all of the predictions it makes. That is, it informs us about whether the model is suitable for the task by showing the number of times it was right out of all attempts.

Accuracy is determined by dividing the number of correct predictions (true positives and true negatives) by the total predictions made, which are the correct and incorrect predictions (true positives, true negatives, false positives, and false negatives).

For example, if a model correctly predicts 90 out of 100 examples, it will have an accuracy of 90%. A higher accuracy typically means a better performance. One should bear in mind, however, that accuracy can sometimes be misleading—especially in imbalanced datasets where one class is more common than other classes. In such cases, a model may appear to be accurate merely because it is skilled at predicting the dominant class but may be awful at predicting the minority class.

So, while accuracy provides a good baseline, it must be considered alongside other metrics such as precision, recall, and the F1-score for a balanced evaluation of model performance.

$$Accuracy = no.\,of\ correct\frac{predictions}{no}.of\ total\ predictions$$

4.2. Precision

Precision is one of the most important evaluation measures that captures the quality of a machine learning model's performance, especially if it concerns the truthfulness of positive predictions. Precision tells us how many instances that the model predicted as positive actually were positive. Another way to say this is that precision answers the following question: "Of all positive predictions made by the model, how many actually were positive?".

In order to calculate precision, you divide the number of true positive predictions by the number of positive instances predicted (true positives and false positives). With high precision, the model is classifying with very few false positive errors—it will only label something as positive when it is extremely confident that it is positive.

Precision becomes particularly important where the cost of a false positive is significant. For example, in spam filters, if a legitimate email message is incorrectly labeled as spam (a false positive), this will result in missed messages of potential value. Similarly, in diagnostic medicine, making a false diagnosis that a patient has a disease when they do not actually have the disease can subject the patient to unnecessary stress and further testing.

But precision has to be calculated along with recall since a model can have high precision and low recall if it is overly conservative in making positive predictions. Precision and recall usually balance each other out in the F1-score, which provides a single metric that accounts for both.

$$Precision = TP + FP$$

4.3. Recall

Recall is a crucial metric that is used to gauge the performance of a machine learning model, more so when the aim is to know how good the model is at identifying positive cases. Recall is all about the ability of the model to capture all respective cases from the dataset. Put another way, it asks, "Of all true positives, what proportion did the model predict accurately?".

To calculate recall, you divide the number of true positive predictions (instances correctly labeled as positive) by the total number of actual positive instances. The latter is the true positives plus the false negatives (instances actually positive but incorrectly predicted as negative).

High recall is a metric indicating that the model is good at detecting most positive instances with minimal false negatives. This is especially attractive in situations where missing a positive instance would be costly or hazardous, e.g., medical diagnosis (e.g., identifying all patients with a disease) or fraud detection.

However, recall should never be considered alone. Although it is important to mark as many positives as possible, this can be at the cost of a lower precision if many negatives are incorrectly marked as positives. Recall is therefore usually measured in combination with precision, and the F1-score is often used to provide a very balanced measure of both scores.

$$Recall = TP + FN$$

4.4. The Confusion Matrix

Table 1 shows the confusion matrix. We achieved an accuracy of 99.67%, which is a good figure and help us make a good decision.

Table 1. Algorithm accuracy.

Classes	True High	True Low	True Medium	Class Precision
Pred. high	68	0	0	100.00%
Pred. low	0	134	0	100.0%
Pred. medium	1	0	115	99.14%
Class recall	98.55%	100.00%	100.00%	

4.5. Specificity

Specificity (also the true negative rate) is one measure that considers how much a model correctly identifies negatives. To be specific, it checks the percentage of true negative instances that were correctly predicted as negative by the model. That is, specificity poses the question, "Out of all true negatives, how many were correctly predicted by the model?".

To calculate specificity, you divide the number of true negatives (instances correctly classified as negative) by the number of false positives (instances incorrectly predicted as positive) and true negatives. Specificity is particularly important in situations where it is highly crucial to avoid incorrectly classifying negative cases as positive. For example, when screening for diseases, high specificity implies that a healthy individual will not be inaccurately labeled with the disease, and therefore, unnecessary treatments or interventions will be avoided.

Though specificity is a desirable measure, it is typically accompanied by sensitivity (or recall), which highlights the degree to which the model detects positive cases. A highly sensitive but low-specificity model can detect numerous negative cases as positive and produce false positives. A highly specific but low-sensitivity model fails to detect any positive cases and produces false negatives. Balancing both measures is crucial to developing an effective model.

$$Specificity = \frac{F_p}{F_p + T_n}$$

4.6. The F-Measure

F-measure or F1-score is a significant measure used in binary classification issues, especially in instances of imbalanced data. It averages the precision and recall into a single measure, providing a balanced approximation of the performance of a model. While precision is concerned with how many of the expected positive instances are truly positive and recall is concerned with how well the model identifies all true positives, the F-measure combines these two metrics to give a more complete idea of the model's effectiveness.

The F-measure is particularly valuable because it avoids the problem of relying on either one of precision or recall, which both are incomplete measures of performance. Where the cost of false positives and false negatives is high (such as in medical diagnoses or detecting fraud), a balanced measure like the F1-score is desirable. For example, in medicine, you want to have a model that recognizes as many positive instances as are recognizable (high recall) without wrongly diagnosing too many healthy individuals as sick (high precision). The F-measure ensures that both are considered.

This is a standard metric in class imbalance scenarios since it discourages the model from focusing on predicting the majority class at the expense of ignoring the minority class. A high F1-score means the model has a high performance both in the detection of positive cases and in avoiding false positives, hence it being a major metric in overall model assessments.

In general, the F-measure cannot be ignored when machine learning with binary classification since it offers a balanced measurement through the inclusion of precision and

recall in terms of one measurement. It has a unique critical role in scenarios where false negatives and false positives are costly, as well as whenever there is imbalanced data.

$$F\ Measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

4.7. Discussion

In this research, we apply a voting system; for better accuracy in the voting system, we apply four different models: decision tree, naïve Bayes, random forest, and KNN. The model in the voting system with the majority vote wins, and the decision goes onto the majority side, which makes us more confident about accurate predictions. We achieved 99.67% accuracy in the end after applying the model. Before this voting system, we applied individual models. We achieved 96.53% with KNN, and with naïve bayes, we achieved 99.47% accuracy; with random forest, we achieved 98.56%; and with the decision tree, we achieved 94.58%, and when we applied the voting system, our results were better.

5. Conclusions

In this research paper, this study looks at how well tools from machine learning, the IoT, and simple light tests can check the quality of oranges. By using the outer appearance of oranges, such as their color and texture, along with what is inside, like tartness and vitamin C content, we yield good results. Of the machine learning methods we tried, naive Bayes worked best, being correct 97.67% of the time. This method is far superior to the traditional ones, offering faster, more reliable checks to both large and small producers. Future work will look to lowering the costs to allow more people to use this technology, improving the citrus chain and making more money. This new method offers good potential for changes to the quality tests in the food industry. We have used a different model to enhance food quality testing.

Author Contributions: A.H. conceived the research idea, designed the methodology, and supervised the study. T.M.A. carried out data collection, preprocessing, and implementation of the experimental models. H. contributed to the analysis, interpretation of results, and preparation of the manuscript draft. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Birla, S.L.; Wang, S.; Tang, J.; Fellman, J.K.; Mattinson, D.S.; Lurie, S. Quality of oranges as influenced by potential radio frequency heat treatments against Mediterranean fruit flies. *Postharvest Biol. Technol.* **2005**, *38*, 66–79. [CrossRef]
2. Bhusal, M.; Bhattarai, D.; Lamsal, M.; Dhital, S.; Adhikari, A. An inception of value chain development of sweet orange; kinesis to food security in the developing countries scenario. *Nepal. Hortic.* **2023**, *17*, 1–9. [CrossRef]
3. Chávez-Mejía, A.C.; Magaña-López, R.; Durán-Álvarez, J.C.; Jiménez-Cisneros, B.E. International Journal of Environment, Agriculture and Biotechnology (IJEAB). *Int. J. Environ. Agric. Biotechnol.* **2019**, *1878*, 1–7. [CrossRef]
4. Krishi Vidyapeeth, M.; Shere, I.D.; Joshi Monali, I.M.; Shere, D.; Monali, J.M. Study of physico-chemical characteristics of sweet orange (*Citrus sinensis*) fruit. *J. Pharmacogn. Phytochem.* **2018**, *7*, 1687–1689. Available online: www.indiastat.com (accessed on 1 January 2025).
5. Mamatova, Z.; Aydın, A. Manas Journal of Agriculture Veterinary and Life Sciences. *Manas J. Agric. Vet. Life Sci.* **2021**, *11*, 100–108.

6. Hassan, H.E.; El-Rahman, A.A.A. Orange quality evaluation during maturity using laser technology. *Misr J. Agric. Eng.* **2014**, *4*, 629–636.
7. Elhosary, M.; El-Metwalli, A.; Derbala, A.; Elsayed, S. Using Image Processing to Evaluate the Quality of Orange Fruits in a Non-Destructive Manner [Unpublished]. 2023. Available online: https://www.researchgate.net/publication/376610236 (accessed on 10 February 2025).
8. Seminara, S.; Bennici, S.; Di Guardo, M.; Caruso, M.; Gentile, A.; La Malfa, S.; Distefano, G. Sweet orange: Evolution, characterization, varieties, and breeding perspectives. *Agriculture* **2023**, *13*, 264. [CrossRef]
9. Apte, S.K.; Patavardhan, P.P. Feature fusion based orange and banana fruit quality analysis with textural image processing. *J. Phys. Conf. Ser.* **2021**, *1911*, 012023. [CrossRef]
10. Quaggio, J.A.; Mattos, D.; Cantarella, H. Fruit yield and quality of sweet oranges affected by nitrogen, phosphorus and potassium fertilization in tropical soils. *Fruits* **2006**, *61*, 293–302. [CrossRef]
11. Orange Quality Data. Available online: https://www.kaggle.com/datasets/shruthiiiee/orange-quality (accessed on 1 February 2025).
12. Babbar, H.; Rani, S.; Masud, M.; Verma, S.; Anand, D.; Jhanjhi, N. Load balancing algorithm for migrating switches in software-defined vehicular networks. *Comput. Mater. Contin.* **2021**, *67*, 1301–1316. [CrossRef]
13. Kok, S.H.; Azween, A.; Jhanjhi, N.Z. Evaluation metric for crypto-ransomware detection using machine learning. *J. Inf. Secur. Appl.* **2020**, *55*, 102646. [CrossRef]
14. Diwaker, C.; Tomar, P.; Solanki, A.; Nayyar, A.; Jhanjhi, N.Z.; Abdullah, A.; Supramaniam, M. A new model for predicting component-based software reliability using soft computing. *IEEE Access* **2019**, *7*, 147191–147203. [CrossRef]
15. Ahmed, S.; Hossain, M.A.; Bhuiyan, M.M.I.; Ray, S.K. A comparative study of machine learning algorithms to predict road accident severity. In Proceedings of the 2021 20th International Conference on Ubiquitous Computing and Communications (IUCC/CIT/DSCI/SmartCNS), London, UK, 20–22 December 2021; pp. 390–397. [CrossRef]
16. Kok, S.H.; Abdullah, A.; Jhanjhi, N.Z.; Supramaniam, M. A review of intrusion detection system using machine learning approach. *Int. J. Eng. Res. Technol.* **2019**, *12*, 8–15.
17. Airehrour, D.; Gutierrez, J.; Ray, S.K. GradeTrust: A secure trust based routing protocol for MANETs. In Proceedings of the 2015 International Telecommunication Networks and Applications Conference (ITNAC), Sydney, NSW, Australia, 18–20 November 2015; pp. 65–70. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.