Classifying Product from their Ingredients Based on various Machine Learning Algorithms

Aharnish Pithva
AU2040022
Ahmedabad University
aharnish.p@ahduni.edu.in

Jevin Jivani
AU2040051
Ahmedabad University
jevin.j@ahduni.edu.in

Astha Bhalodiya
AU2040067
Ahmedabad University
astha.b@ahduni.edu.in

Yug Patel
AU2040181
Ahmedabad University
yug.p2@ahduni.edu.in

***Abstract*—This report investigates the use of machine learning algorithms to classify ingredients into two categories, food or beauty product. Optical Character Recognition (OCR) technology is used to convert images of ingredient labels into text, which is then processed by various algorithms such as Decision Trees, k-Nearest Neighbors, Random Forest, and Support Vector Machines. The performance of each algorithm is evaluated and compared using metrics such as accuracy, precision, recall, and F1-score. The study shows the potential of using OCR and machine learning for ingredient classification in the food and beauty industry.**

***Index Terms*—OCR, Classification, Logistic Regression, k-Nearest Neighbors, Random Forest, Decision Trees, Naive Base, Support Vector Machines**

I. Introduction

Product classification is an important task in many industries, including food and beauty. Classifying products based on their ingredients can be a complex process, but with the help of machine learning algorithms, it can be done accurately and efficiently. In this report, we explore the use of various machine learning algorithms for classifying products based on their ingredients. The ability to accurately classify products from their ingredients has numerous real-life applications. For example, it can be used to identify allergens in food products, which is crucial for individuals with food allergies or intolerances. It can also be used to identify harmful chemicals in beauty products, helping consumers avoid adverse reactions or skin irritations. In addition, accurate product classification can be useful for product development, compliance, and marketing purposes. In our product, user can scan ingredient list from the product they have, and with help of OCR with get text data from photo and will use this list into our trained model and classify accordingly.

In this report, we will explore various machine learning algorithms, including Decision Trees, Random Forest, and Logistic Regression, to classify products based on their ingredients. We will compare the performance of these algorithms and evaluate their accuracy and efficiency in product classification. By the end of this report, we aim to provide insights into the use of machine learning algorithms for product classification based on their ingredients, and their potential applications in various industries.

II. Background/Literature Review

Initially, our project aimed to use OCR to extract text from images and will use NLP to comprehend the context to enable users to ask relevant questions about the image. This technology had potential applications in education, accessibility, and information retrieval. However, we realized that our primary focus was on the deep learning aspect, so we decided to modify our problem statement.

Our new problem statement centers around utilizing an existing OCR scanning model to extract text from images, which will then be fed into our classification model. The primary focus of this project is the classification of ingredients. Additionally, we aim to use a text data set collected from the output of our own images processed by the OCR model.

III. Implementation

A. Data Collection

We attempted to find data from various websites such as Kaggle and Snap, but were unable to find suitable Datasets. We have found some data of ingredients of US market but in our case, we are classifying indian products based on their ingredients, and we wanted to collect data specific to the Indian market. As a result, we decided to create your own dataset by manually collecting ingredient information for products available in the Indian market and websites.

B. Data Cleaning

First we had data in form of a dictionary. Then we took all the ingredients from the different items together. Next, we removed all the unnecessary or irrelevant data. Then we divided all the multi-term ingredients into separate terms. Next, we counted the occurrence of all the ingredients that are present and we got 1110 individual ingredients for food product and 770 individual ingredient for beauty product.

Then, we sorted the ingredients in descending order based on their number of occurrence so we could make a histogram of the top 60 ingredients for both.

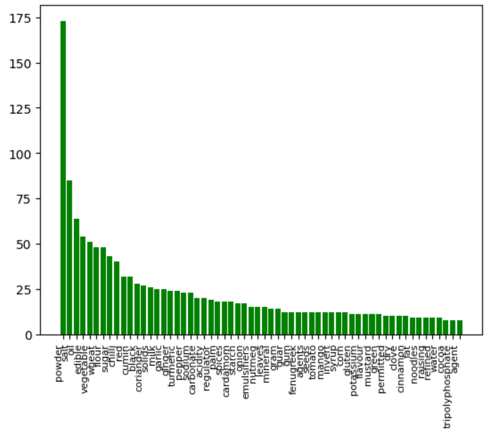

Fig. 1. Top 60 Food Ingredients

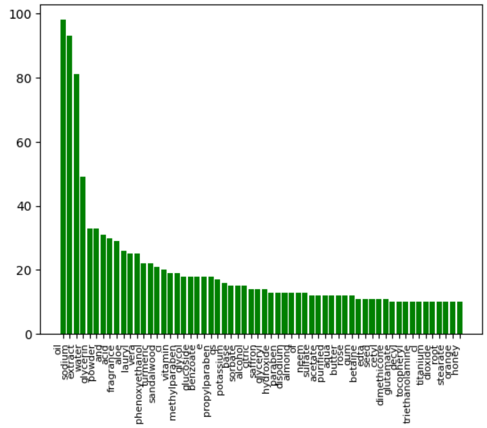

Fig. 2. Top 60 Beauty Products' Ingredients

We analyzed our dataset using Exploratory Data Analysis, a technique used to explore and summarize data to gain insights and detect patterns. As part of our analysis, we created graphs of the top 60 ingredients for food and beauty products used in our data and discovered 7 common words, including Oil, Sodium, Powder, Water, Potassium, Turmeric, and Gum. We also found a total of 106 unique ingredients, bringing our total number of ingredients analyzed to 113. EDA is an important step in any data analysis project as it helps to ensure that data is properly understood and utilized to make informed decisions.

C. One Hot Encoding

One hot encoding is a technique used to represent categorical data as numerical data, which can be used as input to machine learning algorithms. In one hot encoding, each category is represented by a binary vector of 0s and 1s, where the length of the vector is equal to the number of categories.

The advantage of one hot encoding is that it allows machine learning algorithms to treat each category as a separate feature, without imposing any arbitrary numerical order or hierarchy. We have applied One Hot Encoding in ingredient list in our Dataset.

D. Algorithms

We have implemented following machine learning algorithms on our dataset for classification,

1) *Decision Tree:*
2) *Random Forest:*
3) *Logistic Regression:*

RESULTS

E. Selecting Accurate ML Model

1) Accuracy: The accuracy score indicates the correct categorisation of instances in a dataset. Logistic Regression had the highest accuracy of 0.9467, followed by Decision Tree (0.9032) and Random Forest (0.8710). Logistic Regression had the best predictive ability. However, performance indicators such as precision, recall, and F1-score should also be considered before deploying a model. These indicators provide additional information beyond accuracy for assessing model performance.

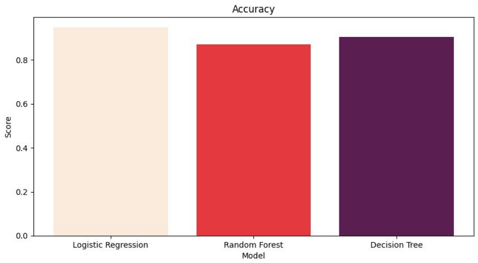

Fig. 3. Accuracy Scores of Machine Learning Algorithms

2) F1-Score: F1 score is a popular metric in machine learning to evaluate classification models. The score ranges from 0 to 1, and higher scores indicate better performance. The most precise model among the three is the "Logistic Regression" model, with a score of 0.97. The "Random Forest" model performs well with a score of 0.87, and the "Decision Tree" model has the lowest performance with a score of 0.90. Overall, based on the F1 scores, the "Logistic Regression" model performs the best, while the other models also perform well but with room for improvement. Remember that these findings may vary depending on the dataset and issue being studied.

3) Precision: A binary classification model's precision is determined by dividing the projected positive cases by the true positive cases. The model with the highest precision score, "Logistic Regression" with a score of 0.98, minimises false positives and can accurately identify many positive cases. Choosing a model with high precision is essential for reliable predictions. Thus, "Logistic Regression" is recommended for applications that require reducing false positives.

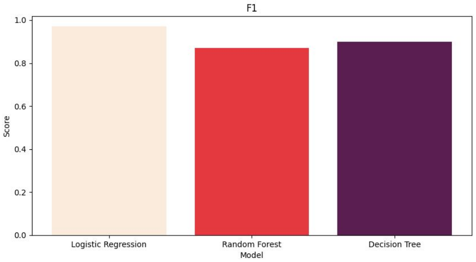

Fig. 4. F1 Scores of Machine Learning Algorithms

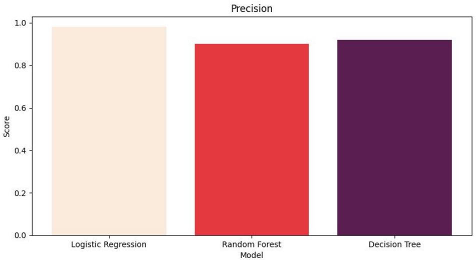

Fig. 5. Precision Scores of Machine Learning Algorithms

4) Recall Rate: The three machine-learning techniques have different recall rates. The logistic regression model has the highest recall rate (0.97), correctly identifying the most positive cases. The decision tree model has a lower recall rate (0.91) which can be considered good, but it identifies fewer positive cases compared to before case. The random forest model has the lowest recall rate (0.86) and does not identify many positive cases.

Therefore, the logistic regression model is the most accurate at detecting positive cases, followed by the decision tree model. The random forest model is the least accurate. These results can be used, along with other evaluation criteria and trade-offs, to select the best machine-learning model for the problem and data at hand.

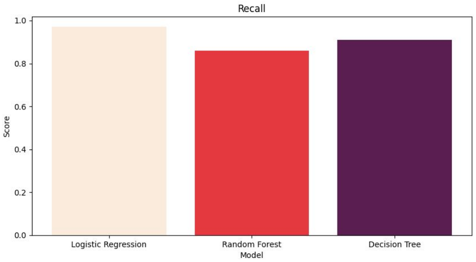

Fig. 6. Recall Scores of Machine Learning Algorithms

F. Checking Underfitting and Overfitting

We tested different train - test split ratios on the logistic regression model and observed the f1 score of the model. The test - train ratio ranged from 0.04 to 1.0. and 0.04 refers to almost all data used in training 1.0 refers to uninitialized weights. We observed that excluding the noise of random initializations each time of testing, a good ratio to prevent overfitting would be 20-80, and the model was less prone to overfitting.

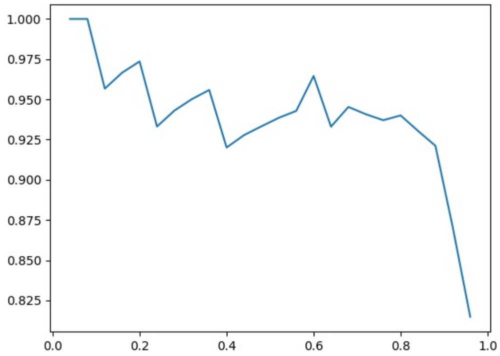

Fig. 7. F1 Score v/s Test/Train Ratio

Conclusion

In summary, the study evaluated the performance of three machine learning models - Logistic Regression, Decision Tree, and Random Forest - based on accuracy, F1-score, precision, and recall rate. Logistic Regression had the highest accuracy, F1-score, and precision, indicating its better predictive ability and ability to minimize false positives. The recall rate was also highest for Logistic Regression, followed by Decision Tree and Random Forest. Further analysis revealed that a train-test split ratio of 20:80 prevented overfitting in the Logistic Regression model. These findings can help in selecting the best machine learning model for a particular problem and dataset. However, it is important to consider other evaluation criteria and trade-offs before deploying a model. It is also essential to note that the results may vary based on the dataset and issue being studied.

Referance

G. [1] Swain, P. H., Hauska, H. (1977). The decision tree classifier: Design and potential. IEEE Transactions on Geoscience Electronics, 15(3), 142-147.

[2] Hosmer Jr, D. W., Lemeshow, S., Sturdivant, R. X. (2013). Applied logistic regression (Vol. 398). John Wiley Sons.

[3] Dasu, T., Johnson, T. (2003). Exploratory data mining and data cleaning. John Wiley Sons.

[4] Pal, M. (2005). Random forest classifier for remote sensing classification. International journal of remote sensing, 26(1), 217-222.//