

School of Science (SofS)

College of Engineering, Environment and Computing (EEC)

**6072CEM Individual Project**

Project Name:

Evaluating model performance on image and numerical data classification tasks

Author's Name: GeRui Ye

Project Supervisor: Manizheh Montazerian

Module Code: 6072CEM

Date: 4/8/2025

**Project Declaration**

I declare that this project was developed by me at Coventry University, and the opinions and conclusions of the research are all from myself unless they are the contents of the cited references, which are all marked with sources.

The paper was not submitted to any other institution; I have submitted the ethics case via ethics coventry.ac.uk The project ID is P184874. And have included the ethics case number above.

Projects without a valid ethics case number will not be accepted and will not be marked.

The code for this project has been uploaded to <https://github.coventry.ac.uk/.> The project can be accessed via the following link: <https://github.coventry.ac.uk/yeg3/6072CEM---Individual-Project.> If you need to view it, please contact the author for access.

Further information is available at [yeg3@coventry.ac.uk](mailto:yeg3@coventry.ac.uk)

Name: GeruiYe

Date: 1/4/2025

Table of Contents

[Abstract 4](#_Toc195042363)

[**1.** **Introduction** 4](#_Toc195042364)

[1.1. Datasets introduction 4](#_Toc195042365)

[**Digital data set aim** 4](#_Toc195042366)

[**Image data set aim** 5](#_Toc195042367)

[**2.** **Literature Review** 6](#_Toc195042368)

[**3.** **Research Methods** 7](#_Toc195042369)

[3.1. Exploratory Data Analysis 7](#_Toc195042370)

[3.2. Image data Analysis 8](#_Toc195042371)

[**4.** **Technical Aspects** 9](#_Toc195042372)

[4.1. User Behavior Prediction Model 9](#_Toc195042373)

[**Optimized data:** 9](#_Toc195042374)

[**Random forest model:** 9](#_Toc195042375)

[**Support Volume Management:** 10](#_Toc195042376)

[**Logistic regression model:** 10](#_Toc195042377)

[4.2. Cross Validation 10](#_Toc195042378)

[4.3. Neural Networks model 11](#_Toc195042379)

[**MLPClassifier(Multi-Layer Perceptron):** 11](#_Toc195042380)

[**Convolutional Neural Network(CNN):** 11](#_Toc195042381)

[4.5. Waste category prediction model 16](#_Toc195042382)

[**5.** **Results testing** 18](#_Toc195042383)

[5.1. Data Evaluation 18](#_Toc195042384)

[5.2. Image model testing 19](#_Toc195042385)

[**6.** **Discussion and Reflection** 20](#_Toc195042386)

[6.1. Technical discussion 20](#_Toc195042387)

[6.2. Issue Reflection 21](#_Toc195042388)

[6.3. Risk Management 21](#_Toc195042389)

[**7.** **Conclusion** 21](#_Toc195042390)

[**BIBLIOGRAPHY** 23](#_Toc195042391)

# Abstract

Artificial intelligence (AI) is currently one of society's most critical research areas; implementing artificial intelligence requires training models. To achieve AI, training suitable models based on large amounts of data is essential. Therefore, having high-quality datasets is crucial. Machine learning is often regarded as a primary technical approach in this process. Several machine learning models are suitable for classification tasks and regression tasks. Examples include traditional machine models: Logistic Regression, Support Vector Machine, Random Forest, deep learning models, Feedforward Neural Networks, Convolutional Neural Networks (CNN) —the main differences in the model assumptions, computational complexity, applicable scenarios, etc. In addition, each model has its characteristics for dealing with different problems and is suitable for dealing with different AI applications.

In addition, the types of data that need to be processed are more varied, dealing with common numeric data, as well as Audio Datasets, image datasets, and text datasets; these datasets are not suitable for every kind of model to deal with these problems;

Therefore, this project uses two datasets from UCI, one purely numerical and the other image data, to construct a machine learning model of the classification problem, observe the differences in the machine model's process in processing these two different data types, and explore the practical significance of the model's construction according to the Literature Review.

**Research Question**

How to build machine learning models suitable for purely numerical data, and build machine learning models ideal for input image data; How to use them for practical applications?

1. **Introduction**
   1. Datasets introduction

**Digital data set aim**

The dataset is obtained from UCI [1](Online Shoppers Purchasing Intention Dataset C. Sakar, 2018). It includes 17 features, such as Visitor Type, Operating Systems, Browser, Region, Traffic Type, Page Value, and Special Day, for 12,330 instances. Within the dataset, 84.5% (10,422) of the samples are negative (i.e., no purchase), while the remaining 1,908 are positive (i.e., purchase made). Hence, the data are imbalanced. When applying this dataset, balancing methods must be used to handle the data; otherwise, overfitting or underfitting may occur.

Here is the dataset information and some of the values.

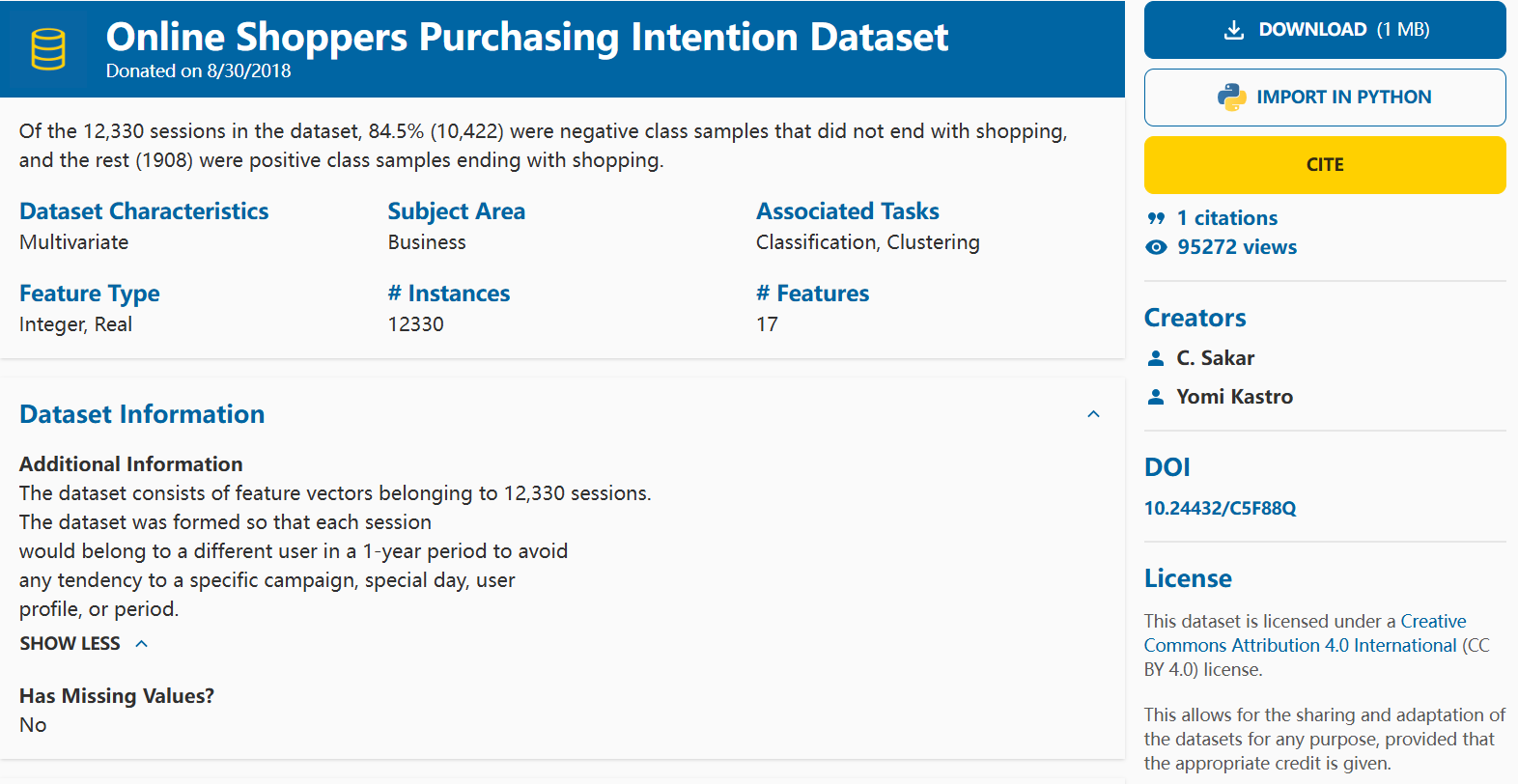


Fig.

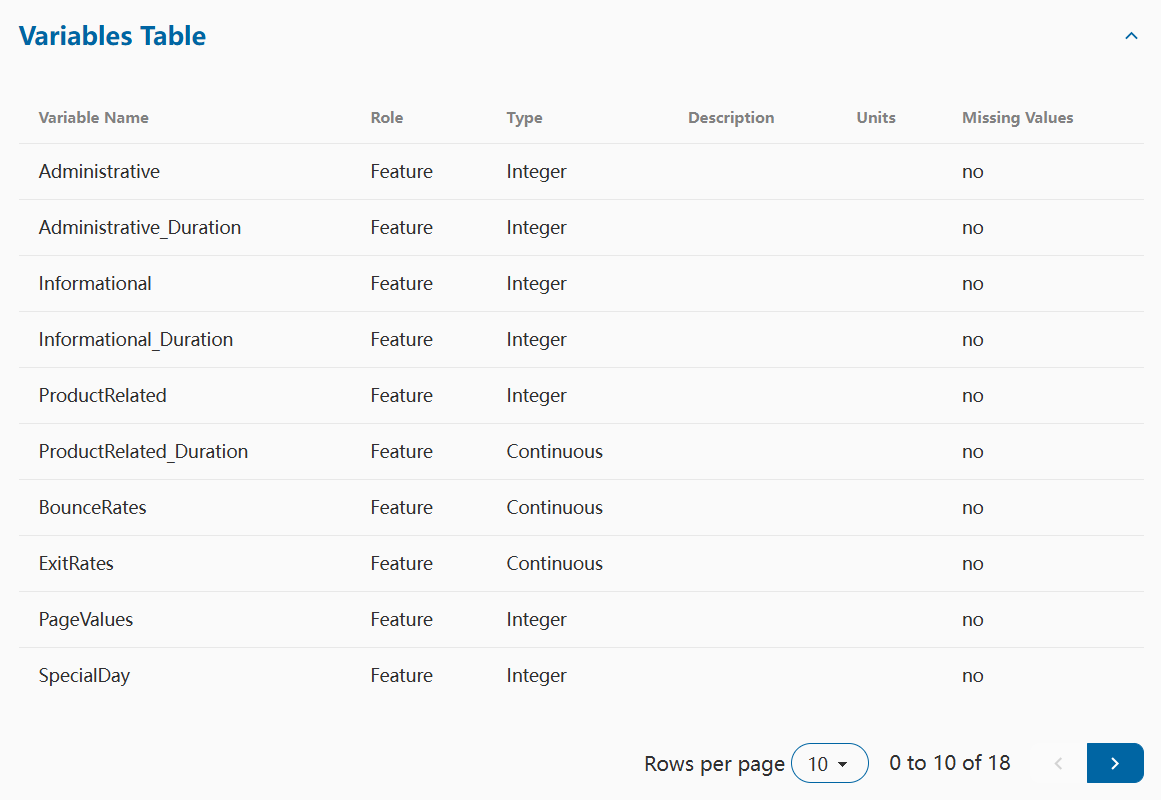


Fig.

This project will use traditional machine learning models and deep learning models on this dataset to construct multiple models in a way that the arithmetic process of each model is recorded in detail, and their results are processed and analyzed. The main objectives are to use this dataset to accomplish the following tasks: (1) Test the performance of each model trained through this dataset. (2) This paper will also analyze the impact of the models on the prediction of e-commerce platforms and user behaviour. (3) This project will review the literature on AI applications, observe their modelling approaches, and consider their strengths and weaknesses from a critical perspective during the research.

**Image data set aim**

The image dataset used is RealWaste from the UCI website, which contains 461 images of Cardboard, 411 images of Food Organics, 420 images of Glass, 790 images of Metal, 500 images of Paper, and 921 images of Plastic; addition to these images: Miscellaneous Trash, Textile Trash, and Vegetation. Miscellaneous Trash, Textile Trash, and Vegetation images, totalling 4752 images of various categories of trash; resolution: 524x524[6] (Single et al., 2023);

This study uses glass, metal, paper, and plastic to train the model with a total of 2,631 images so that the model can distinguish between these four types of waste images. The documents are displayed:

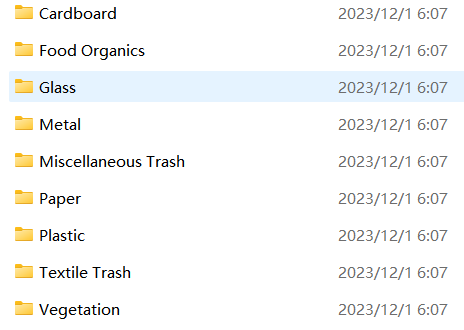


Fig.

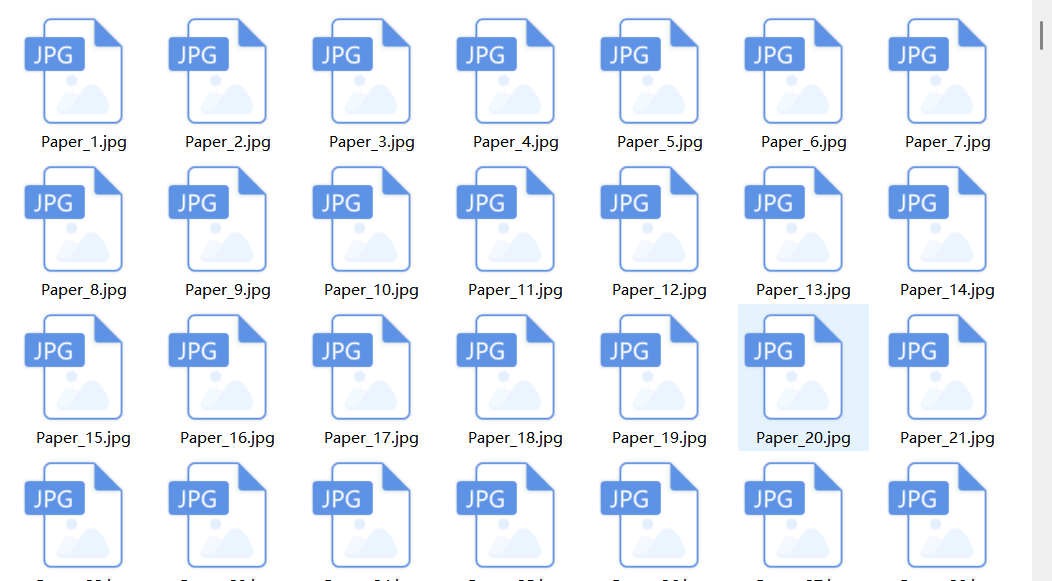


Fig.

This research uses this pure image dataset to construct a CNN machine model that can recognize trash to compare the computational differences with the CNN model constructed from the 'Online Shopper Digital Dataset'. The plan is to accomplish the following tasks using the RealWaste dataset: (1) extract four types of images in the dataset, glass, metal, paper and plastic; construct a model to enable the machine to recognize these four types of objects and display the prediction results through code; (2) compare the computational differences of CNN models when dealing with two-dimensional images and textual data; (3) analyze the practical significance and usefulness of the model.

1. **Literature Review**

This research aims to build models to deal with classification problems, and for classification tasks, standard model-building algorithms include logistic regression, ADA boost, SVM, Bayes, and random forest. For example, Priyanshi Kathuria et al.[4] (2022) used these models to build an ML classification model based on user reviews. Therefore, this study will also take the construction of multiple models. Still, their study used a single dataset for comparison, so in this study, CNN models will be constructed for two datasets of different data types and evaluated to analyse the advantages and disadvantages of various models in handling the problem.

Marion O. Adebiyi et al. (2024)[3] predicted the profitability of e-commerce startups based on their initial funding. They used linear regression models to predict the relevant metrics. Analysing historical data can be effective in providing insights to entrepreneurs and investors. From this point, building models can help the web increase revenue.

The second dataset for this treatment is images, essentially a computer vision problem, which has extensive research in the field, such as recommender systems; Betul AY et al. [15](2019) trained a model using 67,000 images of shoes from a Turkish e-commerce website. It recommends suitable shoes based on user characteristics to enhance the product experience; Smart Furniture and IoT Applications, Lingwei Wang et al. [16](2024) apply computer vision techniques to brilliant furniture, illustrating the role of innovative furniture in the Internet of Things (IoT), as well as other data security and growth issues in smart homes; Classification Systems, Nikita Andriyanov et al.[17] (2023) use computer vision and natural language analysis of posters, and text to classify film labels.

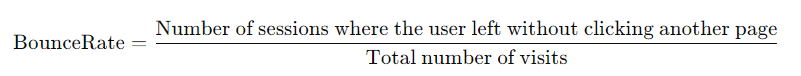
The waste recognition model constructed in this project can achieve the function of waste classification, which can be utilised in recycling stations, dumpsites, etc., to achieve green development.

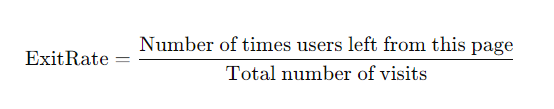
Compared to the above literature, most of them chose to construct models to deal with the problem without multiple levels of comparisons suitable for more kinds of datasets; therefore, the innovation of this research is to construct multiple models based on two different kinds of datasets and judge their performance separately so that people can choose the appropriate model to deal with the corresponding kinds of datasets in the future.

1. **Research Methods**
   1. Exploratory Data Analysis

The digital dataset contains a variety of variables, so the first thing that will be done is to view the data and process the different types of data to make it more suitable to be put into a machine learning model for calculations so that the model is more accurate; according to the analysis of the call to view the data, a total of 12,330 data were found in this dataset, and each of the data has 18 variables; The thirteen variables in which: “Administrative”, “Administrative\_Duration”, “Informational”, “Informational\_Duration”, “ProductRelated”, “ProductRelated\_Duration”, These six data represent the website's statistics on user behaviour, the number of times a user visits and stays on the Administration page, Information page, and product-related pages during the session.

BounceRates indicate that a user enters a page and immediately exits without performing any other actions, and are usually calculated as follows:

  
ExitRates indicates the rate at which this page is the last exit page of a session that is being conducted by the current user, and is usually calculated as follows:



PageValues: the user's contribution to the session, SpecialDay: whether it is a memorable holiday, OperatingSystems: the operating system from which the user is accessing the page, Browser: the browser from which the user is accessing the page, Region: the user's region code, TrafficType: the type of traffic (e.g. advertising, direct access, referral, etc.), The above thirteen types are int64 and float64 types.[5] (Necula, 2023)

There are bool variables Weekend and Revenue. These are converted to 0/1 integers before training the model.

Object character type variables: VisitorType and Month, to maintain the chronological order, use LabelEncoder, which converts the data to integer type.

* 1. Image data Analysis

The RealWaste datasets [6]is the author's collection of nine materials of rubbish images, including metal images, most of which are cans; can have apparent circular features, and other metal images are not folded, with many straight lines; Glass images, most of which are composed of glass bottles or broken glass, the image is characterised by the smoothness of the image is much higher; paper images, composed of newspapers, small bills, envelopes, etc.; Images characterised by: thinness, folding traces of straight lines; plastic images, mostly made up of plastic bottles, tubs, etc.; images with more rounded lines. Some image show：





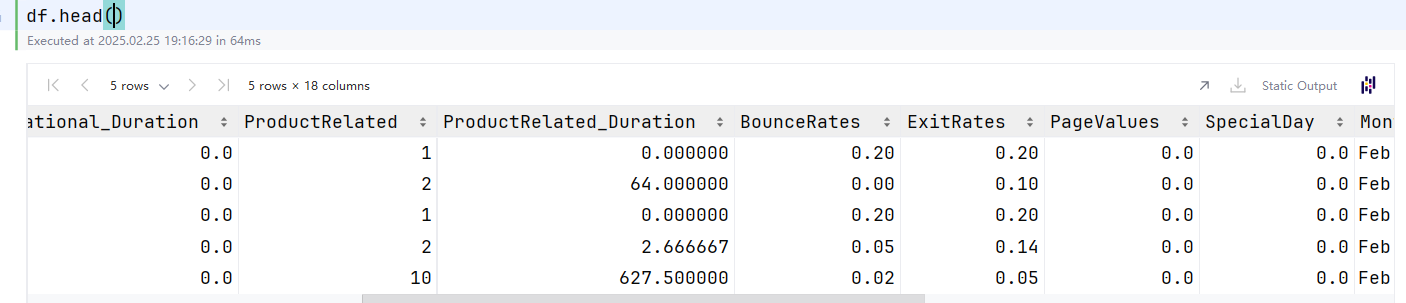
1. **Technical Aspects**
   1. User Behavior Prediction Model

**Optimized data:**

Firstly, StandardScaler() from the scikit-learn library is used to process the data and adjust it to a distribution with a mean of 0 and variance of 1. This method reduces the features of the data that initially have a significant difference to accelerate the convergence of the model, thus realizing the enhancement of the model accuracy and avoiding the substantial impact of certain outliers on the model.

As shown in the figure, the data before processing, after processing, the difference between each data in each row of data will not be particularly large.

Before:



After:

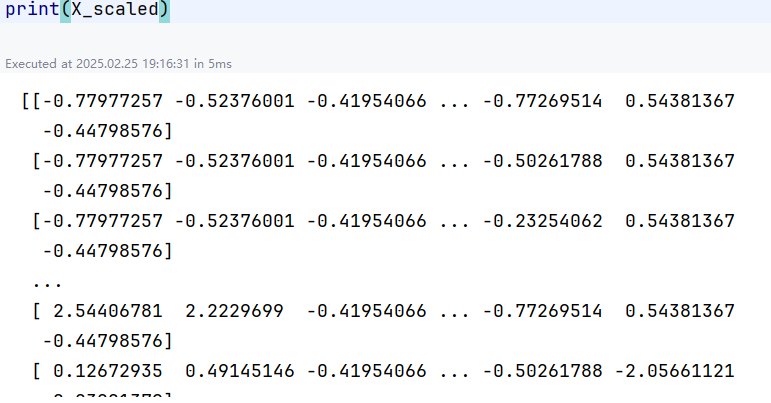
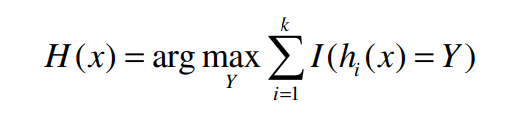


Fig.

**Random forest model:**

In the process of constructing a Random Forest classifier, the model is required to train several decision trees to vote to improve classification performance, so it is necessary to use hyperparameter search space to define the model, n\_estimators represent the number of decision trees, use [100, 200 ] to determine the number of decision trees suitable for the data, max\_depth for defining the depth of the decision tree size, set to [None, 10, 20] to allow the machine to determine the three most appropriate values (None is set the maximum depth), min\_samples\_split is set to [2, 5], i. e., which is how many samples must be included in each decision. The majority voting formula used for random forests in classification problems is:



H(x) is the final predicted category and k is the number of decision trees[11] (Liu et al. 2012), the Original training set, by randomized, is voted to select the final result.

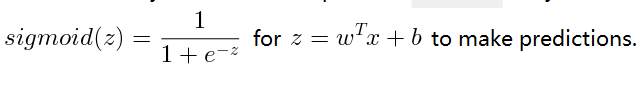
**Support Volume Management:**

To record the duration of each model operation, the project introduces the time module as a timer for the model, from the initialization of the SVM model to the end of the training of the SVM model. Initialization of the SVM algorithm: The RBF kernel function divides the data curves, and a confidence index gives a probability value for the prediction result. Training the model process: The test and training sets are entered into the model using .fit.

**Logistic regression model:**

The logistic regression model handles this classification problem dataset using a sigmoid function calculated on the given input data so that the output value is between [0,1]. The training model process uses the calculated loss function and gradient descent optimization methods to find the best classification boundaries.

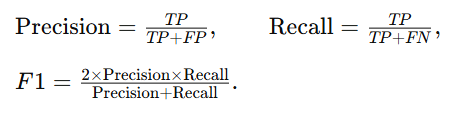
Sigmoid computational formula:



* 1. Cross Validation

After constructing the machine learning model, because no validation sets, so the GridSearchCV (Grid Search + Cross Validation + F1 Indicator) method is used to tune the parameters, this will enable the machine learning model to find the optimal configuration to improve the performance of the model and at the same time ensure the reliability of the final evaluation results, which can improve the generalization performance on the new data; the GridSearchCV will tune the hyper-parameters and learn and train the data to classify it to the best matching neurons[12] (Pirjatullah et al. 2021); The number of cross-validation is chosen to be 5 times, and the use of cross-validation can re-divide the training set and the test set, and then calculate when constructing the model, the code divides 80% into the training set, and the other 20% as the test set, and if only one datasets division is used, then the evaluation results will be dependent on the division of this one time, which will lead to the data to produce a chance component, for example, a particular of data just fit the division of this one time, and the results will be more fragile and reliable.

The F1 score is a comprehensive indicator for judging the model, the mathematical formula:



The purpose of using f1 as a metric is so that the model does not ignore the identification of a few classes, and the F1 score combines to improve precision and recall; using the above method ensures that these model performs evenly across the metrics, rather than appearing to be accurate overall. After the above calculations finish, the .best\_estimator\_ is used to assign the processed model to a variable.

* 1. Neural Networks model

**MLPClassifier(Multi-Layer Perceptron):**

Because the problem is a binary classification problem, so you want to realize the classification problem of FFNN model can be used when the FFNN model can use MLPClassifier to realize the FFNN + automatic partitioning; the model is set to two layers of hidden layers, the first layer is set to 64, and the second layer is set to 32; the activation function of the hidden layer is set to ReLU, using ReLU, it will be the neuron of the hidden layer substitutes for the neuron of f( x) = max(0, x) for the operation (less than zero output 0, more significant than zero linear through), the number of iterations epoch choose 500 times, after 500 times of loop training for data convergence.

After initializing the model, the .fit library is called to automatically perform Backpropagation on the model; backpropagation recalculates the weight gradient by gradient descent, allowing the loss to be gradually reduced.



**Convolutional Neural Network(CNN):**

The CNN model for this data was constructed using a 1D Convolutional Neural Network (1D CNN) from TensorFlow [9]. The module layer classes required to construct the CNN model, including Dense, Input, Flatten, and Adam, were chosen as the optimizer.

The output requirements for Conv1D are (Batch size, Time steps/Features, Channels), three-dimensional values and a channel dimension, so a new dimension is inserted into the original data X\_train and X\_test using np.expand\_dims, axis=2; the latest data after processing becomes a row of 'single-channel' sequence of numbers for the one-dimensional convolutional layer to read smoothly;

Call from Keras[10], Sequential() as a container to build the model, then in the container kind of definition of the input shape, and then set up in Conv1D, 32 filters, so that the CNN model to learn the dataset kind of 32 different feature representations; learning is complete the CNN model needs to call into the Flatten will be the previous layer of features into a one-dimensional vector, and then Using Dense contains 32 neurons fully connected layer using ReLU function for refining, and finally the output layer, the model output layer only needs to contain one neuron, the choice of Sigmoid activation function, the output value will be compressed to 1or0 to achieve the purpose of predicting results.

In addition, the data needs to continue to compile the model. Compile and carry out the cycle of experience; CNN compilation model and traditional machine learning models are similar to the need for optimizer, loss, and metrics as indicators to follow the weights; this time, the use of the Adam optimization algorithm, binary\_crossentropy as binary classification loss function, and accuracy as the evaluation metrics. Finally, the .fit model, the model calculation process for 10 cycles cycle, and the cyclic process model will continue to adjust the parameters, hoping that the loss will reduced and the accuracy increased. cyclic process:

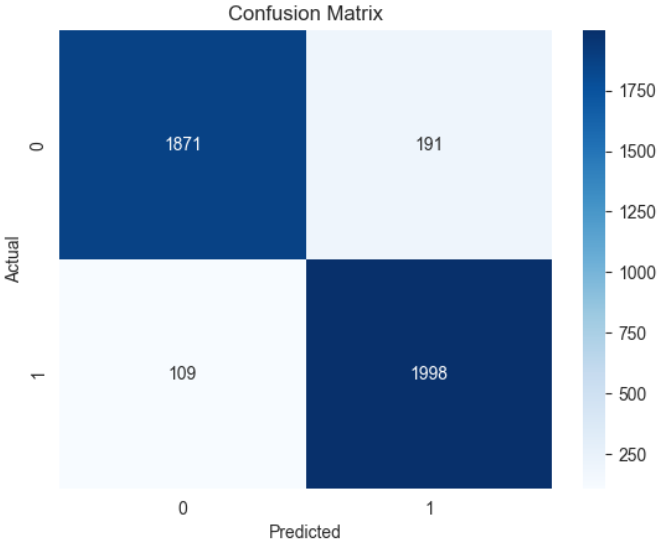


* 1. Model evaluation

**Conventional model:**

Random Forest, SVM, and Logistic Regression print out the model reports for these three models and the confusion matrix to determine the most appropriate model for the binary classification problem. The Random Forest model report and confusion matrix are as follows:

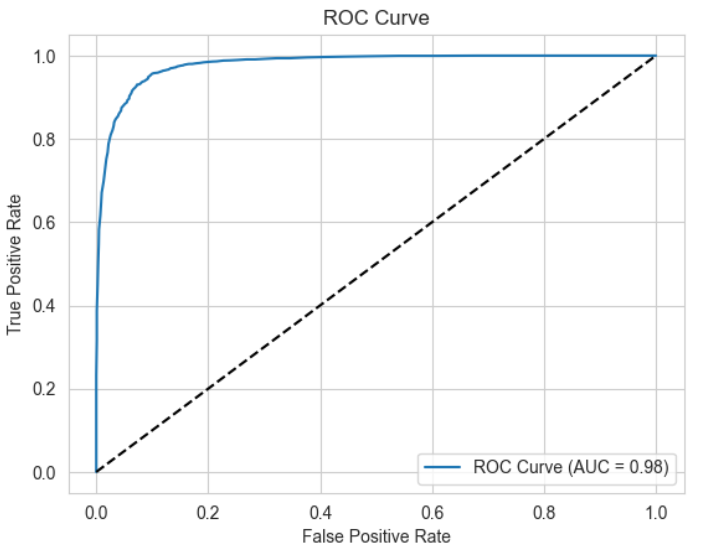




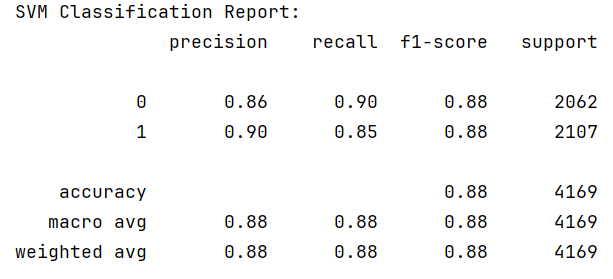
The accuracy of using Random Forest to judge whether the user purchases the current page reaches 92%, while the accuracy of judging whether the user has not made a purchase is 94%. The accuracy of making the purchase is 91%, and out of the 4169 pieces of data, there are 300 pieces of data in which there is a judgment error.

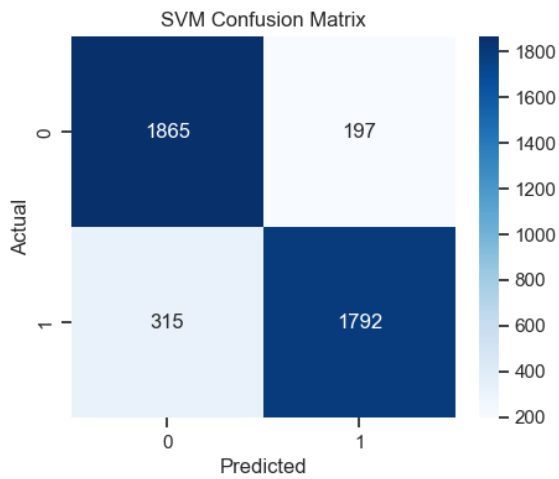
Other data, such as weighted average and macro average, also reached more than 90%, indicating that the comprehensive evaluation of precision, recall, and F1 score is also excellent.

In addition, the Receiver Operating Characteristic (ROC curve) is plotted, which is a two-dimensional curve depicted by pairs of false-positive rate and true-positive rate. The area AUC under this curve can be calculated to assess the model's classification ability [13] (Xu Sun et al., 2014), and the AUC area for the random forest is 0.98.

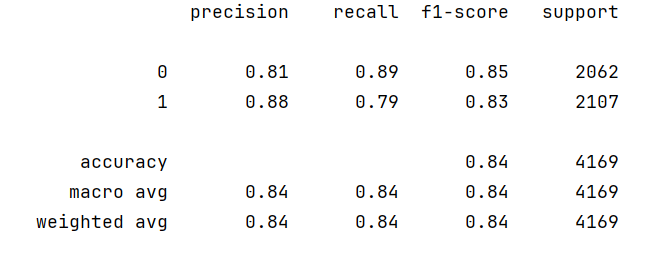


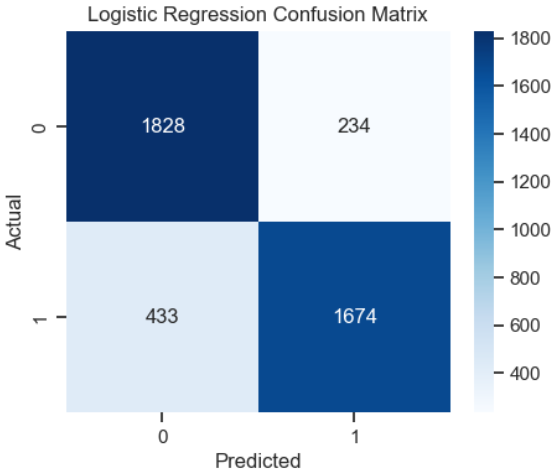
SVM model processing report:





SVM was 86% accurate in determining that a user did not purchase on a web page, 90% correct in making a purchase, and the average of the other metrics was about 88%, with 512 errors in judgment out of 4,169 pieces of data.

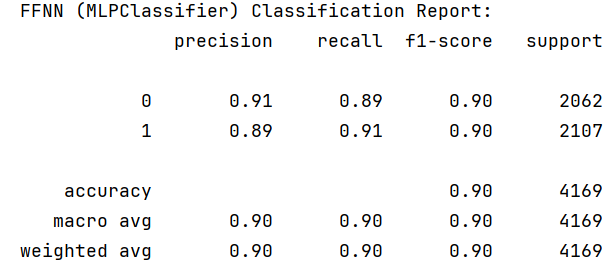


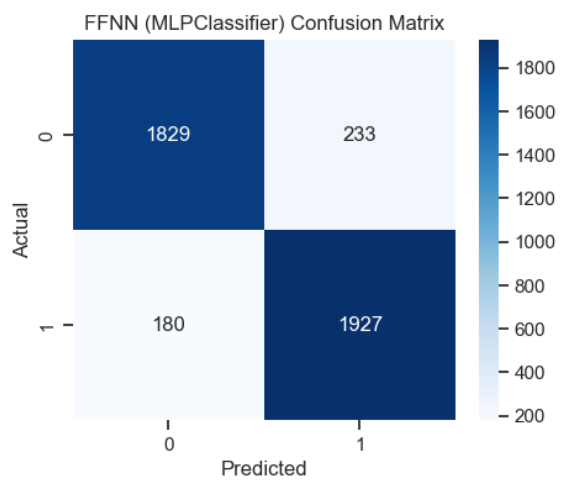


The figure shows the logistic regression model report and the regression matrix; the accuracy of determining that a user bought a product from the site was 88%, the probability of not buying a product was 81%, and the average of the other data was 84%; 234 1s (users who bought a product) were judged as 0 (no purchase), and 433 0s were mistakenly judged as 1.

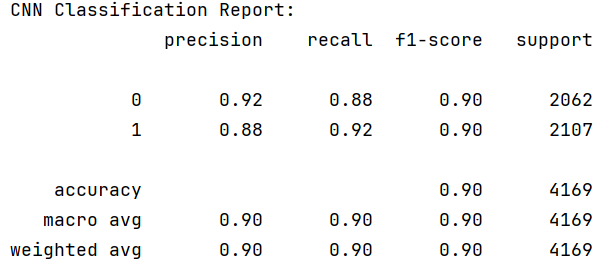
**Neural Networks model summary:**

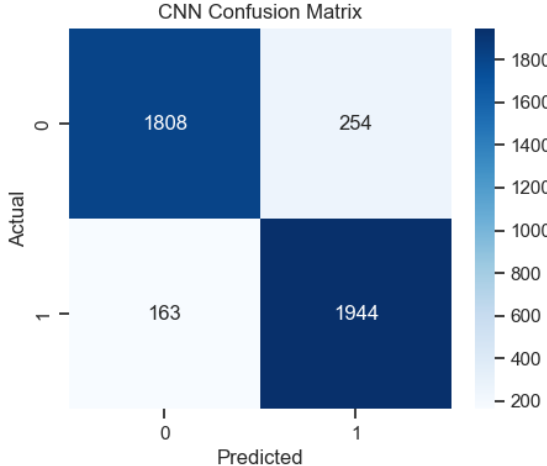
The success rate of this data in training the MLPClassifier model to judge the user's purchasing behaviour is about 90%. The other metrics perform well without overfitting, and out of 4169 pieces of data, there are 413 incorrect judgments. FFNN model reports and confusion matrix plots:





The CNN model processed the purely numerical dataset, and after 10 cycles of computation, the accuracy of judging this binary classification problem was 91% on average. The other metrics performed equally well. According to the confusion matrix, the model had a total of 417 judgment errors.CNN model reports and confusion matrix plots:





**Result comparison:**

When dealing with numerical data, these models' computational process, results, and computation time differ so that the table can reflect the performance of logistic regression, SVM, random forest, MLPClassifier, and CNN models.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Accuracy avg | Error sample | F1-score | Time |
| Random Forest+  Hyperparameter Tuning | 92.5% | 300 | 93% | 4m55s |
| SVM | 88% | 512 | 88% | 53s |
| Logistic Regression | 84.5% | 667 | 84% | 0.03s |
| MLPClassifier | 90% | 413 | 90% | 13s |
| CNN | 90% | 417 | 90% | 17s |

Of the three traditional models, the Random Forest model performed the best in terms of accuracy and f1 scores and was the most suitable model for the dichotomous classification problem; however, this model enabled cross-validation to improve the generalization of the results, and therefore, the training time was the longest of all the models. The logistic regression model is not as accurate as the other models, but at the same time, it has the most straightforward computational rules and the shortest training time. The MLP model and the CNN model perform similarly on this dichotomous classification problem, which suggests that CNN still performs no worse than the other models when dealing with numerical data;

The SVM model has higher complexity and, thus, longer training time. However, it can divide the data into several intervals for secondary computation, e.g., using (CNN+SVM) method for image classification problems.[14]（Banerjee et al. 2023)

* 1. Waste category prediction model

In the model, processing image data, processing digital data, and processing text data have different points, but recognizing image data still needs to process image data and find the digital relationship between the images so as to give the computer recognition and achieve the effect of computer vision.

This project utilizes the 'RealWaste' image dataset, which contains images of garbage made of different materials. Four categories of garbage images, namely glass, metal, paper, and plastic, are used to construct a CNN model to test its accuracy in recognizing garbage made of these four different materials. The accuracy of the model in recognizing these four different materials of garbage is tested, and finally, the computing process will be compared with the process of processing digital data.

To deal with the image problem, we first need to enhance the image data and preprocessing, in short, randomly adjust the angle of the image, so that the machine reads the image does not look so "positive", the purpose is to improve the generalization ability of the model, the training set using ImageDataGenerator from the Keras library, its The specific content includes: (1) image normalization, edit the rescale to scale the pixel values to [0,1], to prevent the gradient from disappearing or exploding; (2) select the rotation\_range as 40, rotate the image randomly by 40 ° to allow the model to randomly identify the angle; (3) set the image to randomly pan horizontally or vertically; as well as flipping the image, to prevent the image from relying on a specific direction; (4) set the image to randomly translate horizontally or vertically; as well as flip the image, to prevent the image from relying on a particular direction dependence; (4) set random brightness [0.7,1.3], and also use 'nearest' to fill the picture after panning.

To deal with the image problem, we first need to enhance the image data and preprocessing; in short, randomly adjust the angle of the image, so Also divide the training and test sets need to be defined: same image size; give the model 32 images at a time; and the training set needs shuffle=False to cancel the order. And the validation set to replace K-folding.

According to the processing, the number of the training set, validator, and test set images：

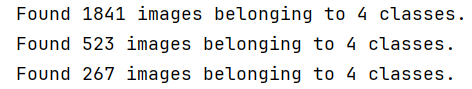


Fig.

The model construction is similar to the processing of numerical models; the input layer uses Conv2D to compile three convolutional blocks and set three colour channels; the fully connected layer setting to 128, and an anti-overfitting design is .dropout(0.4) prevents remembering too many details of the picture; the output layer set to softmax for the multiclassification problem, model structure:

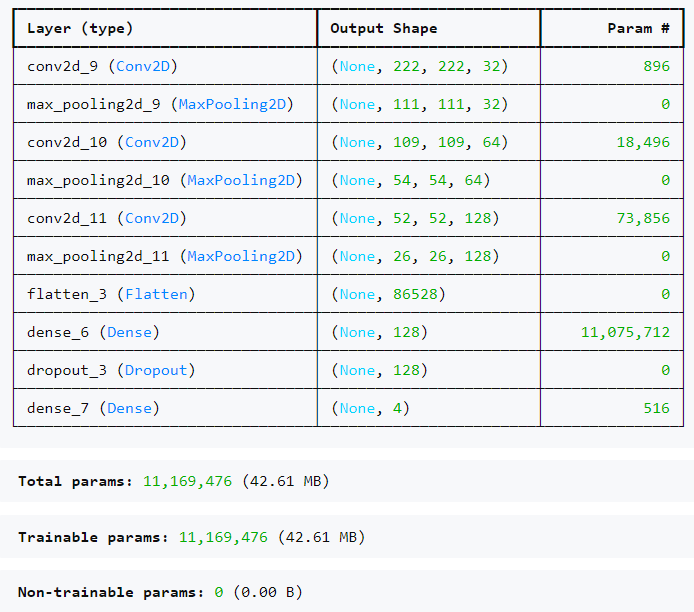
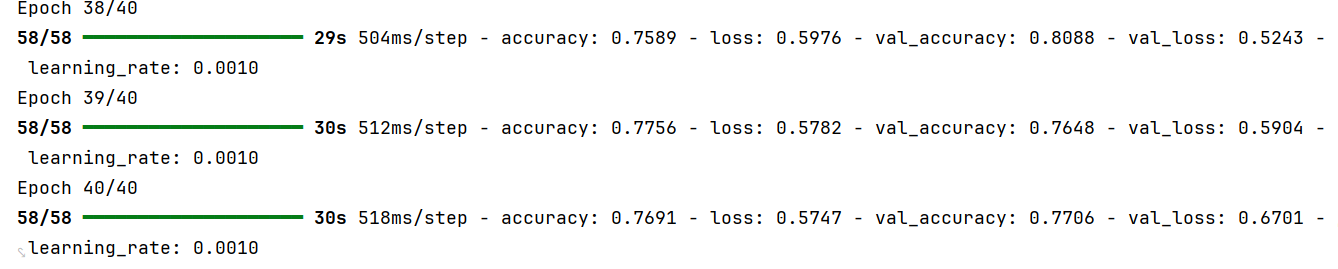


Fig.

According to the picture, we can observe that the processed images are about 200, the parameters to be processed are the first layer (3x3x3+1)\*32 = 896, and the parameters of the fully connected layer: 86,528x128 + 128 = 11,075,712; finally, we use the fully connected structure of Flatten() + Dense().

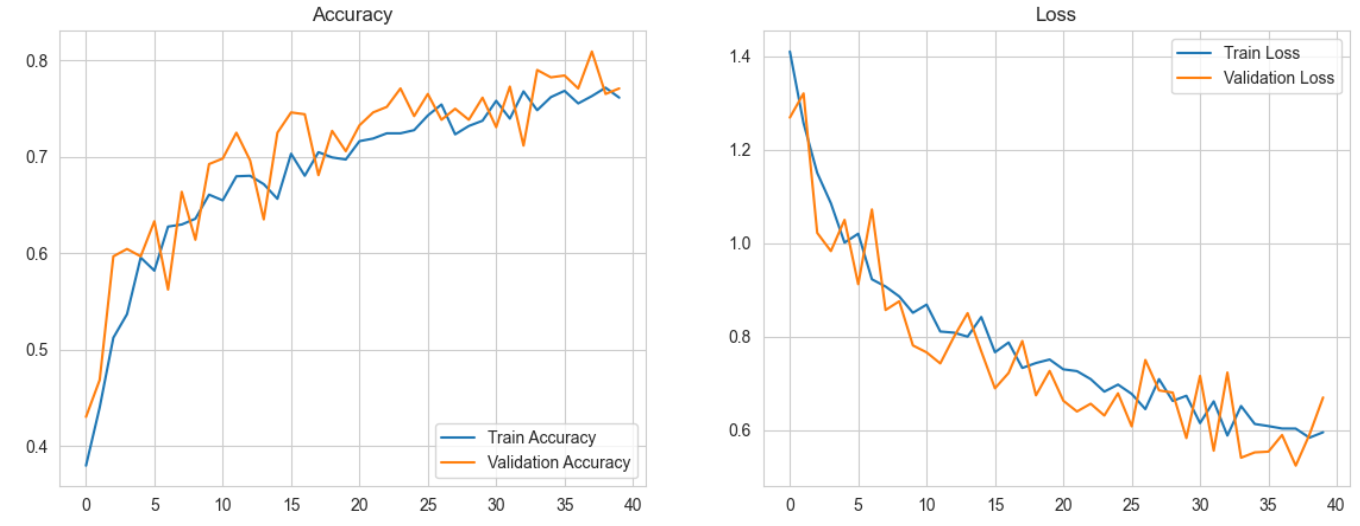
The project compiles the model using Adam, and the loss function is categorical\_crossentropy, which is suitable for multi-classification problems. EarlyStopping is set up to halt training if the val\_loss does not decrease for 10 val\_losses. ReduceLROnPlateau is used to monitor the loss exponent, and if it does not reduce the learning rate, the model has the control to decrease the learning rate, ensuring its adaptability and stability.

Setting the epoch to 40 allows the die to be trained for 40 validations, the training process:



It can be observed that as the number of times increases during the training process, the Accuracy of the test set and the validation set is steadily increasing, and the Loss of both sides is also steadily decreasing. There is no abnormality in the training process.

Compile the variation curves of Accuracy and Loss using matplotlib:



1. **Results testing**
   1. Data Evaluation

For online shopping data, you can use the dataset's ‘.feature\_importances\_’ to calculate the weight of each row of data and use visualisation methods to represent the data. It can be seen that ‘PageValues’ has the most significant influence on the decision to buy or not to buy and can be focused on when dealing with the problem; the weight of each column affecting the user's decision to buy:

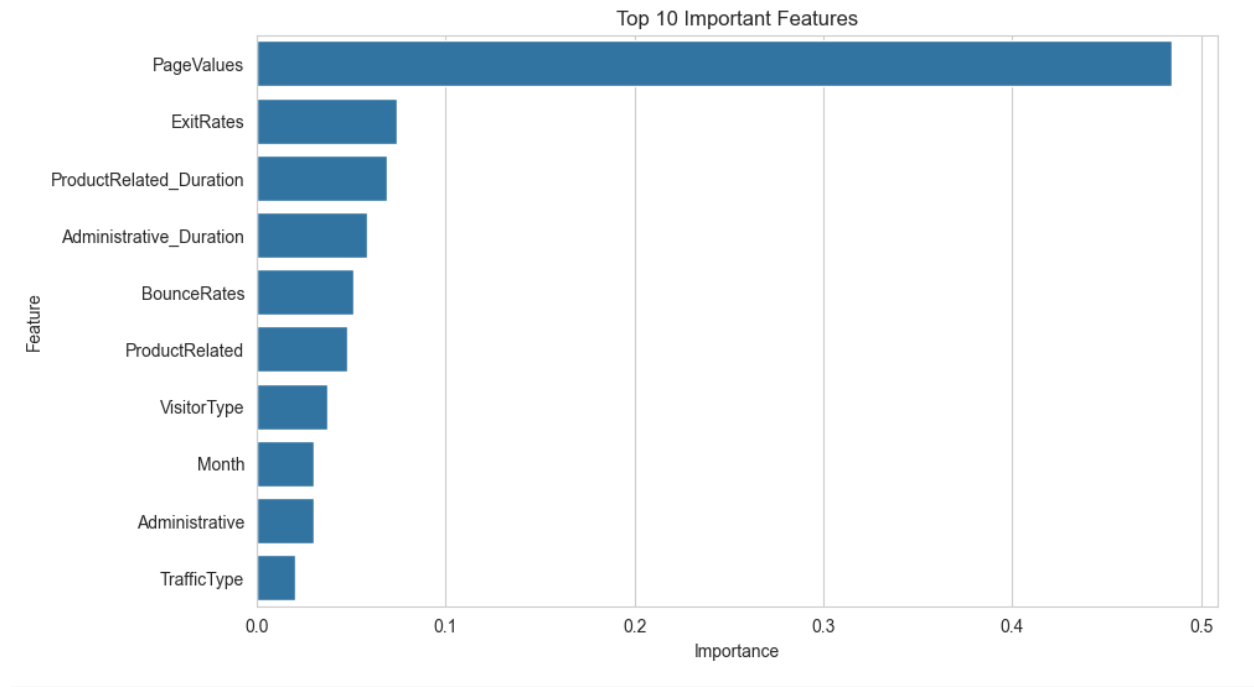
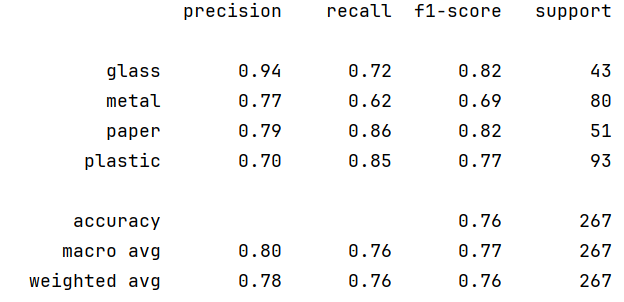


Fig.

* 1. Image model testing

The image recognition model's average accuracy in judging the garbage of four types of materials (glass, metal, paper, plastic) is 76%, with the highest success rate of 94% for glass. However, the number of verified images for glass is only 43, potentially inflating the accuracy. For metal, the model's accuracy is about 77% based on 80 images, and it is expected that the model can correctly classify metal with even higher accuracy in the future.



Subsequently, four pictures were randomly selected, respectively, these four types of materials, to test whether the model can run; first of all, the loaded picture is scaled to fit the model to read (224 \* 224) and then the data is converted to NumPy, followed by the transformation of dimensionality and normalization [0-1] uniform brightness; subsequently, calculate the probability that the picture belongs to one of these four types of material [0.92, 0.03, 0.04, 0.01] The highest probability of 0.92 is judged to be the object in which the value is located;

Example of test results:





1. **Discussion and Reflection**
   1. Technical discussion

Python 3.12 is used as the core development language for the model. It has a rich library for machine learning and data analysis and is highly stable. As Ross Smith et al. [2] (2016) demonstrated, the advantages of Python compiled code are closely related to computationally intensive workloads, which is why it is the best candidate language for vision-based object recognition.

And use Jupyter Notebook as the primary interface for development. The code is divided into modules, and the output of each module is observed independently, so it is convenient to quickly find the problem module for debugging. It can also be used to explain the code by marking cells for human review.

The CNN models constructed from the two datasets show that in terms of purely numerical datasets, the models are processed faster, with the longest time usually not exceeding 5 min in 1w pieces of data; however, when dealing with image datasets, the models are more complex in structure, with longer processes, such as layering the images and normalising the pixel points, etc., and constructing the models usually takes 20 min. The accuracy is not as high as that of the models.

After constructing the initial model again, to achieve the best performance of the model, it is necessary to analyse the impact of these adjustments on the classification performance by adjusting the hyperparameters in the code or non-model related variables such as input feature selection several times, and observing the performance of the evaluation results of the optimised model before determining the optimal model configuration scheme for the dataset.

* 1. Issue Reflection

When dealing with image data, the authors have used images with only one object in them; if the future dataset is less than optimal, for example, an image that contains plastic as well as metal, it is possible to mislead the model, so a picture that includes two or more objects can be outputted with the object at the highest according to the probability of [59,40,0.5,0.5], which indicates the presence of metal as well as the image Plastic.

This numerical data does not contain missing values. Still, if missing values are encountered, The median and mean can fill in the missing values, facilitating the smooth construction of the model.

* 1. Risk Management

In addition, the accuracy of machine learning models, as well as to achieve the level of effective differentiation of data, but the success rate is not 100%, can not rely entirely on machine judgment; so, if machine learning is applied to real life such as garbage sorting, there are bound to be a small number of errors in judgment; thus, people need to add protection, such as adding a final layer of manual screening, monitoring the judgment of incorrect recycling goods.

The purpose is to prevent hazardous substances or non-recyclable objects from entering the recycling channel and causing damage to machinery or personal safety. Examples of errors in judgment:



Fig.

1. **Conclusion**

Based on the digital dataset obtained from the collected information on the operation of the user's interactive web, it is possible to determine whether the user purchased the shopping site based on the user's “information activity” and “product-related activity”, the corresponding “information duration “ and “product-related durations” and whether the purchase occurred on a weekend, to finally determine whether the user purchased the shopping site; this measurement is used to determine whether the user ultimately purchased the product or not, which allows for a better and more detailed assessment of the e-commerce site's profitability. It can also help identify popular products to increase revenue or remove unpopular, less purchased products to reduce inventory and allocate resources more efficiently. Enhance personalised recommendations for e-commerce and improve the experience, Helps the web filter the role of users

Picture classification model From the practical value point of view, then it can achieve the function of computer vision, used in a single program, can reach a small part of the goal, such as rubbish classification, film classification and other work; in the future can be for the training model to input more, a massive amount of picture data, can be constructed, for example, the road recognition, robot eyeballs, to help the community to complete a more complex task.

# **BIBLIOGRAPHY**

1. Sakar, C. & Kastro, Y. (2018). Online Shoppers Purchasing Intention Dataset [Dataset]. UCI Machine Learning Repository. <https://doi.org/10.24432/C5F88Q.>
2. Smith, R. (2016). Performance of MPI codes written in Python with NumPy and mpi4py. 2016 6th Workshop on Python for High-Performance and Scientific Computing (PyHPC) (pp. 45–51). <https://doi.org/10.1109/PyHPC.2016.010>
3. Adebiyi, M. O., Ajayi, S. A., Olaniyan, D., Olaniyan, J., & Kikelomo, D. (2024). Start-up capital estimation and profitability prediction for e-commerce start-ups using machine learning. In 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG) (pp. 1–7). Omu-Aran, Nigeria. <https://doi.org/10.1109/SEB4SDG60871.2024.10630338>
4. Kathuria, P., Sethi, P., & Negi, R. (2022). Sentiment analysis on E-commerce reviews and ratings using ML & NLP models to understand consumer behavior. In 2022 International Conference on Recent Trends in Microelectronics, Automation, Computing and Communications Systems (ICMACC) (pp. 1–5). IEEE. <https://doi.org/10.1109/ICMACC54824.2022.10093674>
5. Necula, S. C. (2023). Exploring the impact of time spent reading product information on e-commerce websites: A machine learning approach to analyze consumer behavior. Behavioral Sciences, 13(6), 439.
6. Single, S., Iranmanesh, S., & Raad, R. (2023). RealWaste [Dataset]. UCI Machine Learning Repository. <https://doi.org/10.24432/C5SS4G.>
7. Matplotlib Development Team. (2023). Matplotlib: Visualization with Python. Retrieved June 5, 2024, from <https://matplotlib.org>
8. Scikit-learn developers. (n.d.). Scikit-learn: Machine learning in Python. <https://scikit-learn.org>
9. TensorFlow. (2024). TensorFlow: An end-to-end open source machine learning platform. Retrieved February 19, 2025, from <https://www.tensorflow.org>
10. Keras. (2024). Keras documentation. Retrieved February 19, 2025, from <https://keras.io>
11. Liu, Y., Wang, Y., & Zhang, J. (2012, September). New machine learning algorithm: Random forest. In International conference on information computing and applications (pp. 246-252). Berlin, Heidelberg: Springer Berlin Heidelberg.
12. Pirjatullah, D., Kartini, D. T., Nugrahadi, M., Muliadi, & Farmadi, A. (2021). Hyperparameter tuning using GridSearchCV on the comparison of the activation function of the ELM method to the classification of pneumonia in toddlers. Proceedings of the 4th International Conference of Computer and Informatics Engineering (IC2IE), Depok, Indonesia, 390–395. <https://doi.org/10.1109/IC2IE53219.2021.9649207>
13. Sun, X., & Xu, W. (2014). Fast implementation of DeLong’s algorithm for comparing the areas under correlated receiver operating characteristic curves. IEEE Signal Processing Letters, 21(11), 1389–1393. <https://doi.org/10.1109/LSP.2014.2337313>
14. Banerjee, D., Kukreja, V., Hariharan, S., Jain, V., & Vats, S. (2023). Mango disease detection system using CNN and SVM: A comparative study with traditional diagnostic methods. In Proceedings of the 2023 World Conference on Communication & Computing (WCONF) (pp. 1–6). IEEE. <https://doi.org/10.1109/WCONF58270.2023.10235224>
15. Ay, B., Aydın, G., Koyun, Z., & Demir, M. (2019). A visual similarity recommendation system using generative adversarial networks. In 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML) (pp. 44-48). Istanbul, Turkey. <https://doi.org/10.1109/Deep-ML.2019.00017>
16. Wang, L., Zhu, F., & Wang, H. (2024). Research and implementation of computer vision and Internet of Things in smart homes. Proceedings of the 2024 IEEE 7th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 587–590. <https://doi.org/10.1109/ITNEC60942.2024.10733225>
17. Andriyanov, N., & Dementev, V. (2023). Combining computer vision and word processing to classify film genres. In 2023 25th International Conference on Digital Signal Processing and its Applications (DSPA) (pp. 1-5). IEEE. <https://doi.org/10.1109/DSPA57594.2023.10113362>