Machine learning and neural network

Food Classification & coffee sales forecasting

Supervised by: Dr. Ali Alrodan

Group Members: Obada Wajeeh Hamdan 0227167
Renad Ayman Naser 0222145

Second Semester
2025

1.0 Introduction

1.1 Overview of the problem

Classification task:
Image classification is an important task in computer vision with many real-world applications. In this project we used the food industry, the goal is to classify food images into one of the predefined categories using Machine Learning models trained on features extracted from a convolutional neural network (CNN).

Time series task:
Accurate sales forecasting is essential for operational planning and strategic decision-making, especially in the food and beverage industry. This project focuses on time series forecasting of coffee sales using real-world point-of-sale data

1.2 Objectives

Classification task:

- Train and evaluate different machine learning models (SVM, MLP, Random Forest) for image classification.
- Evaluate the models using metrics such as accuracy, precision, recall, F1-score, and AUC-ROC.
- Compare each model performance with the chosen task and identify the most effective approach.

Time series task:

- Train and Develop separate implementations for Echo State Network (ESN), Long Short-Term Memory (LSTM), and Bidirectional LSTM (Bi-LSTM).
- Compute forecasting accuracy using metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-squared (R^2).
- Compare each model performance with the chosen task and identify the most effective approach.

1.3 Chosen datasets

Classification task:
The dataset used in this project is the **Food11 Image Dataset**, available on Kaggle. It contains images of food items categorized into 11 different classes. For this project, a subset of 8 food classes (Bread, Dairy product, Dessert, Egg, Fried food, Meat, Rice, and Seafood) was selected to focus the classification task and reduce computational overhead

Time series task:
The dataset used in this project is the **Coffee Sales Datase**t from Kaggle. It contains several columns; date, datetime (timestamp), cash_type, card, money(amount), coffee_name(product bought).

2.0 Datasets

Classification task:
[Food11 Image Dataset]
Chosen classes: Bread, Dairy product, Dessert, Egg, Fried food, Meat, Rice, and Seafood

Preprocessing Steps:

1. Class filtering: only 8 classes were selected from the original 11 classes dataset.
2. Image Limiting: limiting images for only 300 images per class to ensure balanced class distribution .
3. Training set : 8 classes * 300 images = 2400 images
   Validation set : 8 classes * 300 images = 2400 images
   Testing set: 8 classes * 300 images = 2400 images
4. Images resized to 224x224 pixels
5. Preprocessed using preprocess_input from keras.applications.resnet50
6. Data loading performed using ImageDataGenerator for consistency

Time series task:
[Coffee Sales Dataset]

Preprocessing Steps:

1. Datetime conversion: Timestamps were converted to datetime format and resampled to daily frequency.
2. Daily total sales were calculated by summing the 'Amount' column
3. Days with no sales were filled with zeros to maintain continuity in the time series.
4. Normalization: Sales values were scaled using Min-Max normalization to ensure model stability during training.
5. The data was divided into training (80%) and testing (20%) sets based on time.

3.0 Methodology

Classification task:

1. Feature extraction
   A ResNet50 model pre-trained on ImageNet was used as a feature extractor. The model was used without the top classification layers, features were extracted using global average pooling on the convolutional base.This approach leverages transfer learning to provide robust feature vectors for classification.

2. Classification Models
   Several machine learning models were trained on the extracted features.
   A. Support Vector Machine (SVM)
      Linear Kernel is used with probability Estimation Enabled. SVMs are effective for high-dimensional data and often perform well with image-based features.

B. Multi-layer Perceptron (MLP)
   Two hidden layers with 256 and 128 neurons, activation is default (ReLU for hidden layers), maximum iterations of 300 iterations and random speed is set at 42 for reproducibility. MLPs can learn complex decision boundaries on top of deep features

C. Random Forest
   Number of estimators is 100 and random speed is set at 42, ensemble models like Random Forests can capture non-linear relationships and reduce overfitting.

3. Calorie Estimation and External Testing (using SVM model)
To extend the utility of the food classification system, a calorie estimation module was added. This involved mapping each predicted class to an average calorie value per 100g based on nutritional references. The mapping was hard coded using a dictionary, allowing the system to estimate the caloric content once the food class was identified.Additionally, the trained model was tested using food images sourced from Google Images. These images were not part of the Food11 dataset and served as a way to evaluate how well the model generalized to real-world, unseen data.

Time series task:
We implemented and compared the following models for time series prediction.

1. Echo state network(ESN): with hyperparameters
   n_inputs=window (window is 30 days)
   n_outputs=1
   random_state=42
   n_reservoir=600
   sparsity=0.2
   noise=0.001
   spectral_radius=0.8
   input_scaling= 0.7

2. LSTM
   LSTM is a type of RNN that addresses the vanishing gradient problem and captures long-term dependencies.
   Hyperparameters: Hidden units=64, number of layers=2 , Dropout=0.3, learning rate=0.001, loss function=MSEloss, Epochs=50, batch size=32

3. BiLSTM
   Same as LSTM but the bidirectional version: nn.LSTM(..., bidirectional=True)

4.0 Experiment Setup

4.1 environment

Classification task:
- Language: Python using VS code and Pycharm
- Frameworks: TensorFlow, Keras, scikit-learn, matplotlib, seaborn.
- Dataset: Food11 (Kaggle)

Time series task:
- Language: Python using VS code and Pycharm
- Libraries: NumPy, Pandas, Scikit-learn, Matplotlib, PyTorch, PyESN
- Dataset: coffee sales (Kaggle)

4.2 tools

Classification task:
- KaggleHub for dataset
- TQDM for progress visualization
- Matplotlib / Seaborn for plotting
- Scikit-learn for traditional ML models and metrics

Time series task:
- KaggleHub for dataset
- Matplotlib for plotting

4.3 Training configurations and justification

Classification task

- ResNet50 is a robust model for transfer learning and efficient for feature extraction.
- Classical ML models offer quick training and interpretation when using deep features.
- Limiting the dataset size ensures fast experimentation without sacrificing performance analysis.

Time series task:

- Models were trained on normalized daily sales data.

- LSTM training was done with early stopping based on validation loss to prevent overfitting.

- ESN used a fixed reservoir and was trained using ridge regression.

5.0 Results & Analysis

5.1 Results

Classification task:

Model	Val Accuracy	Test Accuracy	Precision	Recall	F1-Score	AUC-ROC
SVM	0.8064	0.8370	0.8377	0.8370	0.8367	0.9831
MLP	0.8206	0.8459	0.8461	0.8459	0.8455	0.9842
Random Forest	0.7598	0.7779	0.7780	0.7779	0.7754	0.9646

Confusion Matrices

SVM

MLP

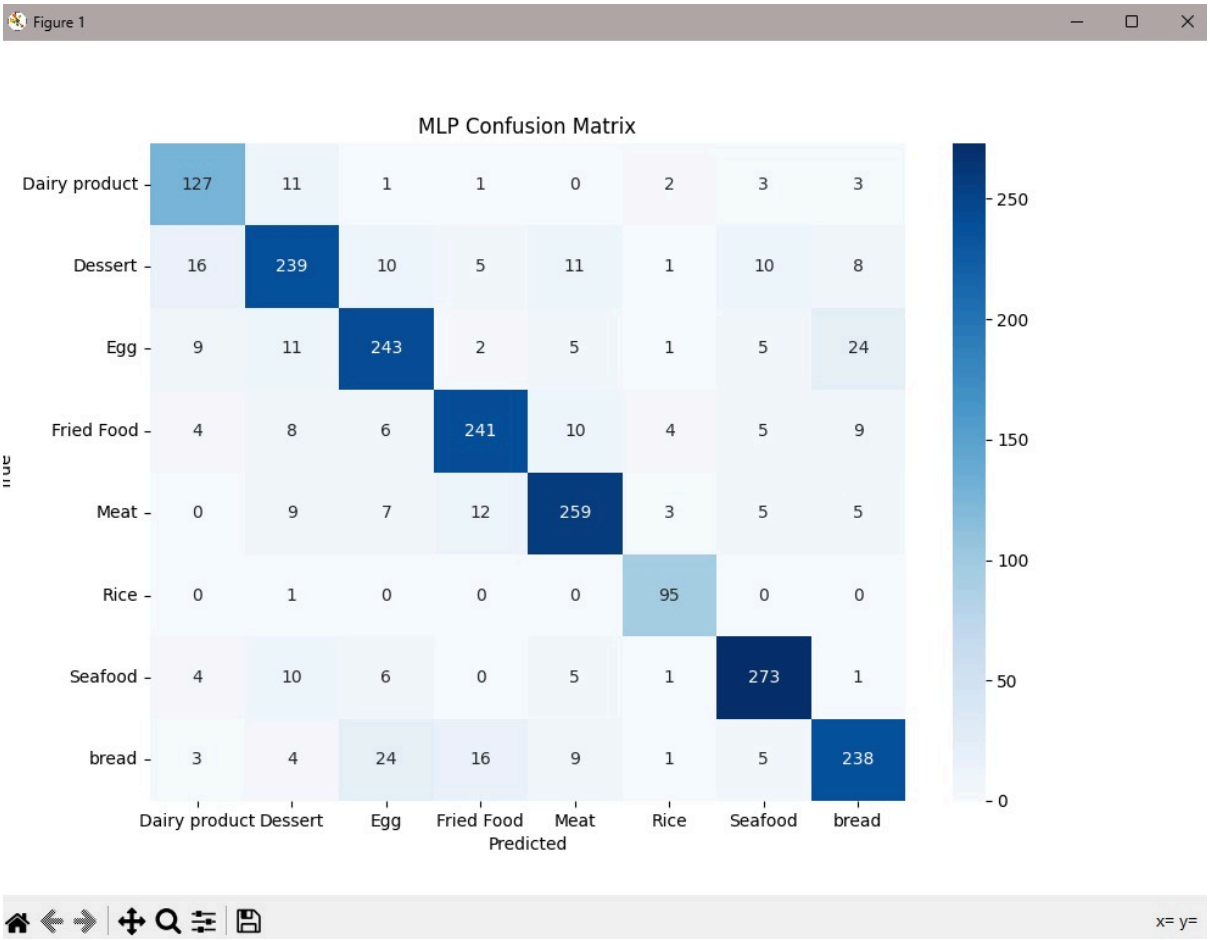


Figure 1
MLP Confusion Matrix
Dairy product
Dessert
Egg
Fried Food
Meat
Rice
Seafood
bread
True
Predicted
x= y=

Random Forest

Time series task:

Model	Train MSE	RMSE	MAE	R^2
ESN	0.0067	45.6842	36.3227	0.8165
LSTM	0.6981	0.8355	0.6899	0.8314
BiLSTM	0.8834	0.9399	0.7781	0.7866

ESN

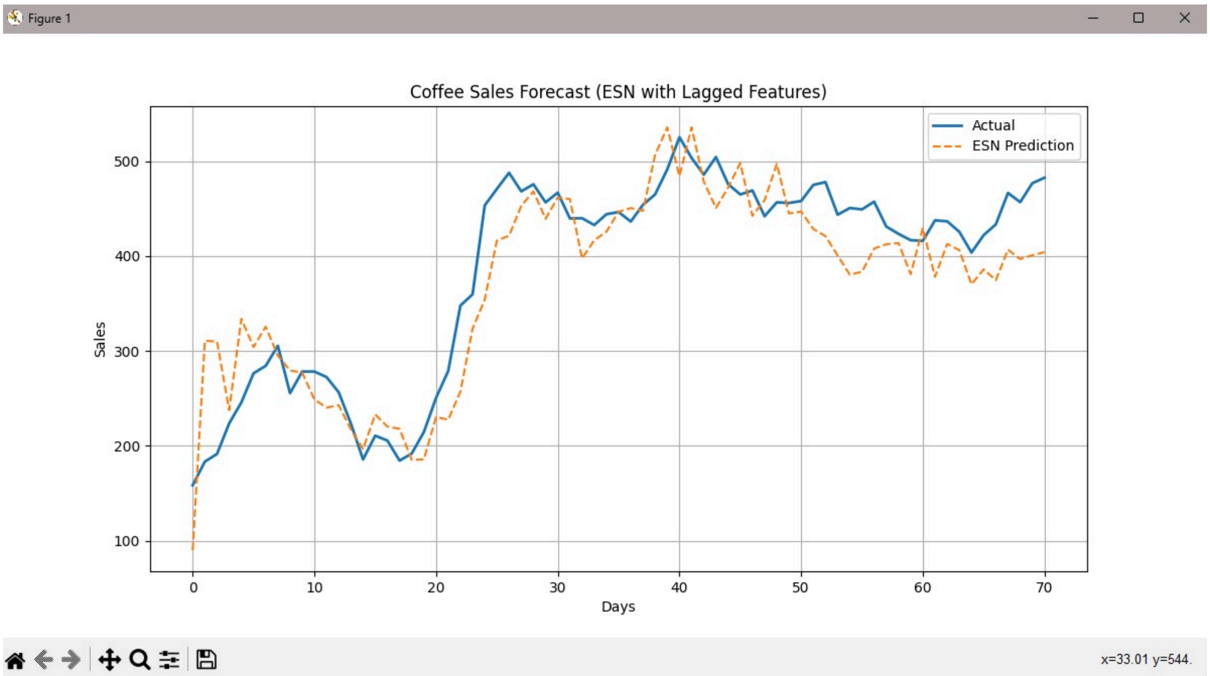


Figure 1
Coffee Sales Forecast (ESN with Lagged Features)
Actual
ESN Prediction
Sales
Days
100
200
300
400
500
0
10
20
30
40
50
60
70
x=33.01 y=544.

LSTM

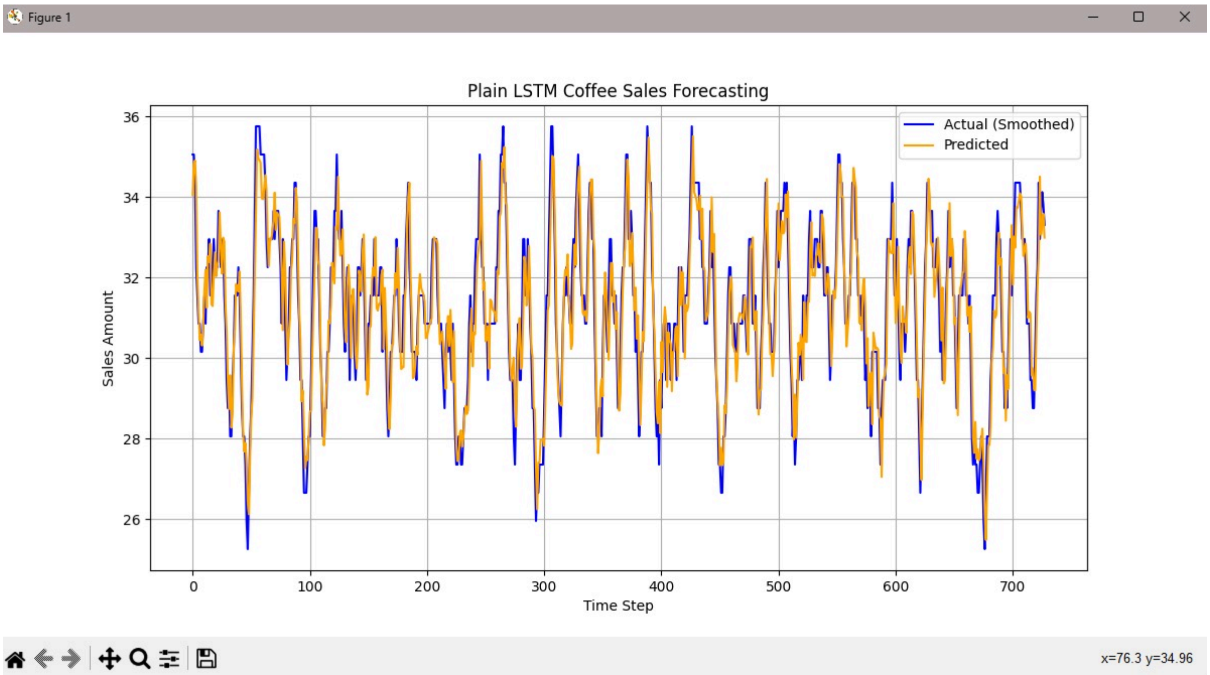
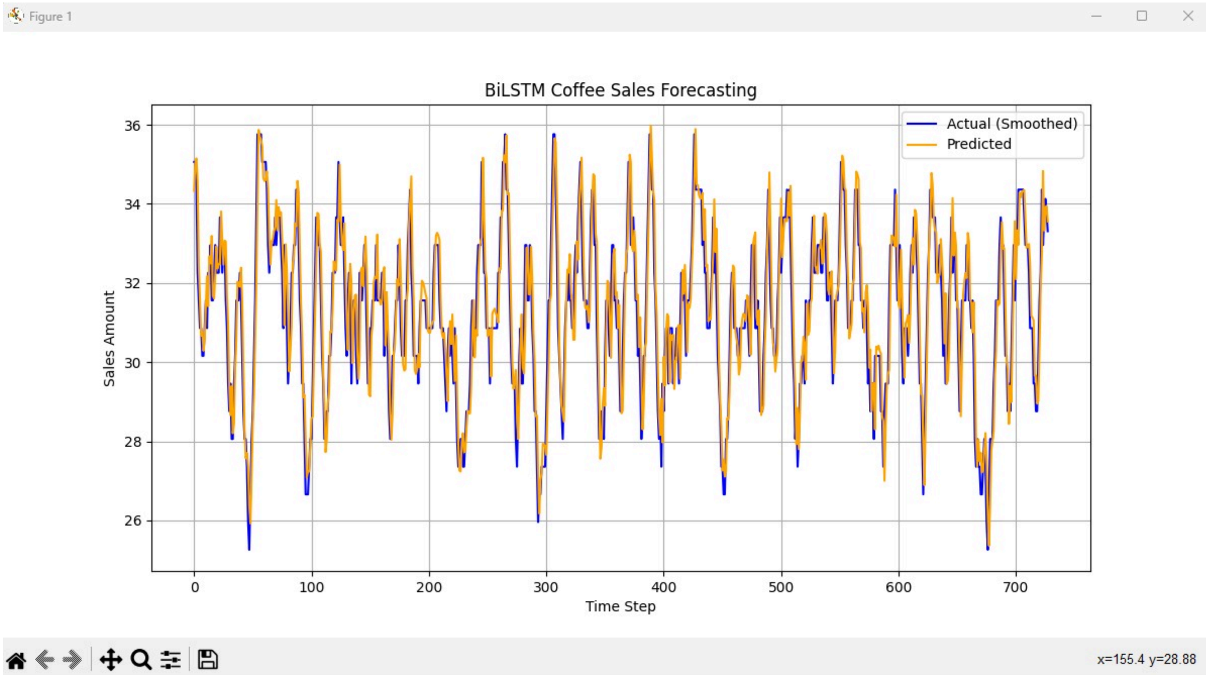


Figure 1
Plain LSTM Coffee Sales Forecasting
Actual (Smoothed)
Predicted
Sales Amount
Time Step
26
28
30
32
34
36
0
100
200
300
400
500
600
700
x=76.3 y=34.96

BiLSTM



5.2 Analysis

Classification task

- All models perform well, with MLP slightly outperforming others in validation.
- SVM and MLP achieved similar F1-scores and AUC-ROC.
- Random Forest slightly underperformed, possibly due to the curse of dimensionality in high-dimensional feature space.

Time series task:

- LSTM outperformed ESN in both MAE and RMSE, suggesting better overall accuracy.
- However, ESN required significantly less training time and achieved reasonable accuracy, making it attractive for low-resource or real-time applications.
- Both models captured the general trend and seasonality, though LSTM handled noisy fluctuations better.

6.0 Conclusion & Future Work

Classification task

Extracting features using ResNet50 enhances the performance of traditional machine learning classifiers. SVM and MLP have competitive results making them the most suitable for real-world lightweight applications. For future work we look forward to exploring using all 11 classes of this dataset and using linear regression for calorie estimation.

Time series task:
In this project, we forecasted daily coffee sales using time series models — Echo State Network and LSTM. LSTM achieved superior prediction accuracy. ESN, though slightly less accurate, proved efficient and effective for rapid deployment. For future work we look forward to exploring hybrid models combining LSTM with ESN.

7.0 References

Classification task:
Kaggle dataset: https://www.kaggle.com/datasets/trolukovich/food11-image-dataset
ResNet50, Scikit-learn, TensorFlow Keras, ImageDataGenerator, etc

Time series task:
Kaggle dataset: https://www.kaggle.com/datasets/ihelon/coffee-sales
PyESN, Pytorch, scikit-learn, matplotlib, etc