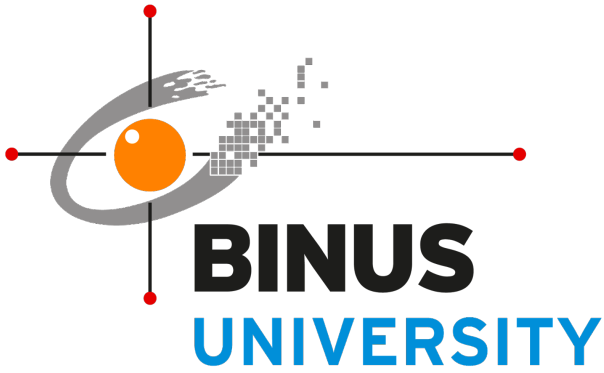
Odd Semester (2023/2024)



BINUS UNIVERSITY

Final Project Report: "Food Recognition"

Student Information:	Surname:	Given Name:	Student ID Number:
1.	Jacques	Ferdinand	2501982600
2.	Joe Claudio	Alexandro	2501963160
3.	Wirawan Ichwan	Wilbert	2501963186
4.	Valerianus Budiman	Nicholas	2502055596

Course Code : COMP6065001 **Course Name** : Artificial Intelligence

Class : L5AC **Lecturer** : Zhandos Yessenbayev, B.Sc., M.Sc., Ph.D.

Type of Assignments : Final Project Report

Due Date : 17 January 2024

The assignment should meet the below requirements.

1. Assignment (hard copy) is required to be submitted on clean paper, and (soft copy) as per the lecturer's instructions.
2. Soft copy assignment also requires the signed (hardcopy) submission of this form, which automatically validates the softcopy submission.
3. The above information is complete and legible.
4. Compiled pages are firmly stapled.
5. Assignment has been copied (soft copy and hard copy) for each student ahead of the submission.

Plagiarism/Cheating

BiNus International seriously regards all forms of plagiarism, cheating, and collusion as academic offenses which may result in severe penalties, including loss/drop of marks, course/class discontinuity, and other possible penalties executed by the university. Please refer to the related course syllabus for further information.

Declaration of Originality

By signing this assignment, I understand, accept, and consent to BiNus International's terms and policy on plagiarism. Herewith I declare that the work contained in this assignment is my own work and has not been submitted for the use of assessment in another course or class, except where this has been notified and accepted in advance.

Signature of Student:

1. Ferdinand Jacques
2. Wilbert Wirawan Ichwan
3. Alexandro Joe Claudio
4. Nicholas Valerianus Budiman

I. Problem description

Our project will be to develop an AI-powered image classifier specialized in recognizing and categorizing various types of food items from images. The objective is to create an AI model that is capable of accurately identifying different foods depicted in images, aiding in automated food recognition and classification.

Challenges:

- Diverse Food Categories: The classifier needs to identify and categorize a wide range of food items from different cuisines, cultures, and presentation styles. This diversity poses a challenge due to variations in color, shape, texture, and composition of food items.
- Hardware Challenges: Due to the nature of AI models needing a strong GPU, CPU, and RAM our project faces lots of hardware challenges. Our original dataset experienced multiple instances of crashes and errors before we decided to split it off into a smaller data set consisting of less classes and fewer images.
- Real-Time Performance: To be practical for various applications, the classifier should achieve a balance between accuracy and speed. It needs to deliver prompt and reliable results, especially in scenarios where real-time classification is essential.

The objective is to create a robust deep learning model for accurately recognizing and categorizing various types of food items in images, with a preference for using convolutional neural networks (CNNs). To achieve this goal, a comprehensive dataset is curated and preprocessed, encompassing diverse images of different foods, ensuring a representative collection that spans various cuisines and presentation styles. To enhance the model's resilience against variations in image quality, lighting conditions, and background complexity, data augmentation techniques are implemented. The training process involves fine-tuning the model to attain high accuracy in food classification, with careful consideration for computational efficiency, especially for real-time applications. The evaluation of the model's performance is conducted using metrics such as accuracy to gauge its effectiveness in accurately identifying and categorizing food items in images.

II. Solution features, e.g. algorithms used, special technologies used, user interface, etc.

Our solution starts off with the dataset which we obtained through the following link https://www.kaggle.com/datasets/kmader/food41. We chose this data set as it has a wide array of food classes with each class being relatively deep and containing almost 1000 images. This allows for a diverse yet accurate classification model.

We then moved on to choosing the right algorithm and AI model to use for our dataset, after some consideration we chose to train a google/vit-base-patch16-224-in21k model and using hugging face to assist us in managing our data. Due to time constraints for this project, we chose to make a simple GUI using the TKinter library which lets the users utilize and view the results of the prediction.

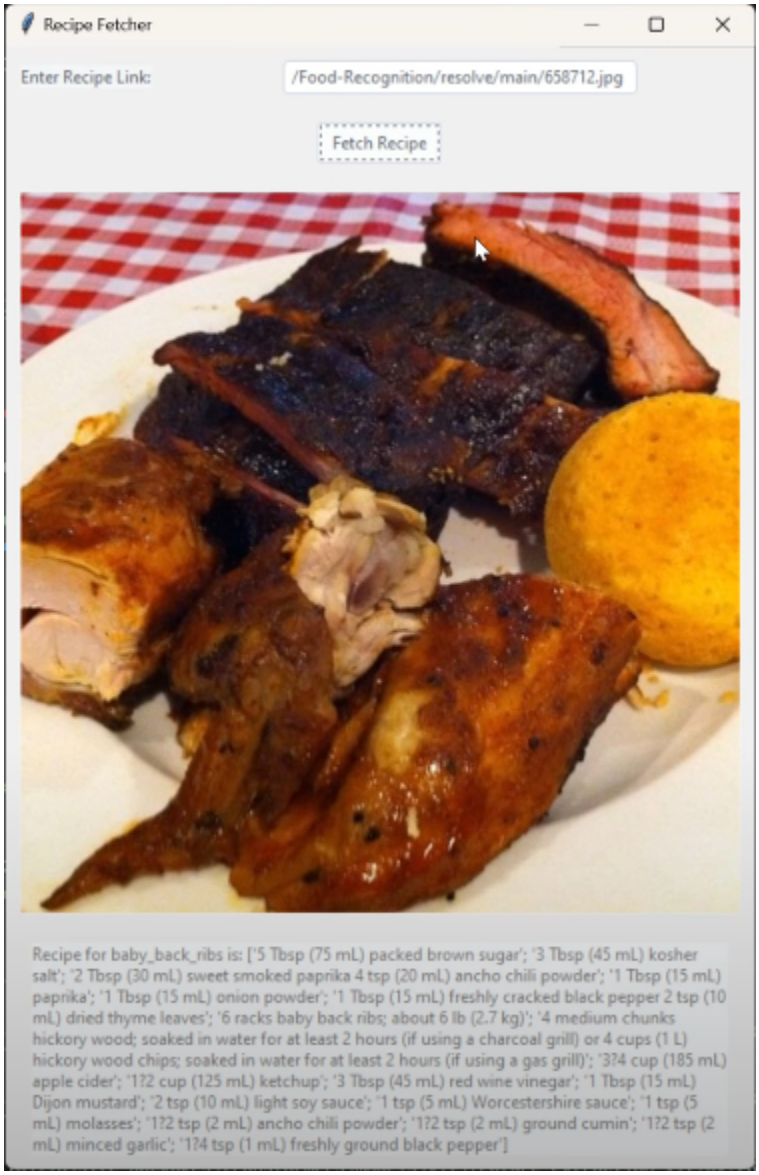
III. Solution design architecture

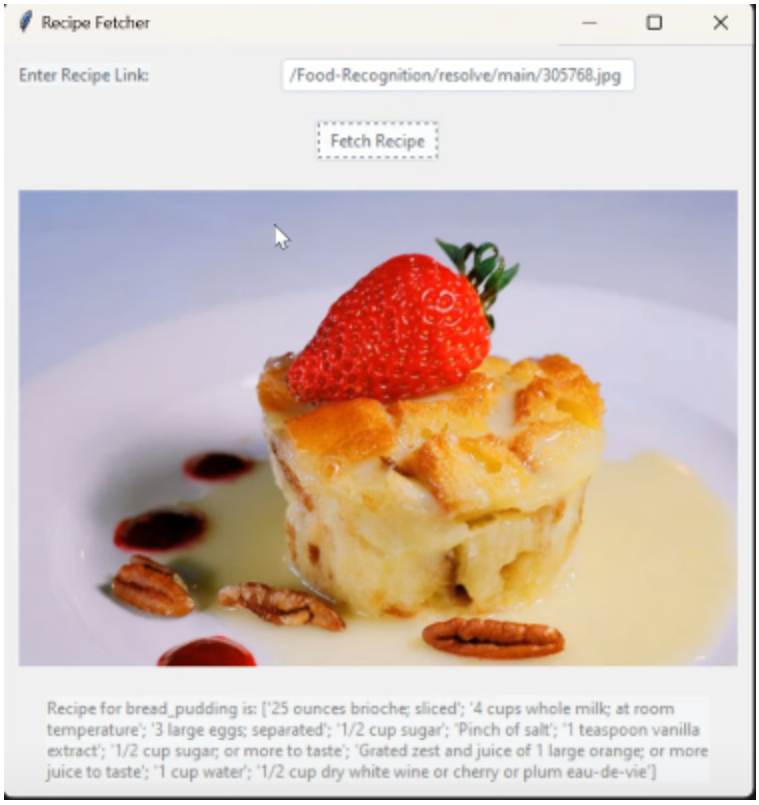
- Data Preparation:
    - Dataset Selection: We choose a suitable dataset for image classification in kaggle.
    - Data Loading: Use PyTorch's data loading utilities (torchvision.datasets, torch.utils.data.Dataset, etc.) to load and preprocess the dataset.
    - Data Augmentation: Cutting down the size of the data to better fit our hardware specs and capabilities
- Model Architecture:
    - Model Selection: We chose google ViT base, Vision Transformer is a transformer encoder model pre trained on a large collection of images in a supervised fashion.
    - Implement Model: Construct the model using google ViT by defining layers, connections, and forward pass logic.
- Training:
    - Loss Function: We used a Trainer from the Transformers library to compute the loss during training.
    - Training Loop: Write the training loop using PyTorch, involving forward propagation, loss calculation, backward propagation (gradient calculation), and optimizer step.
    - Fine Tuning : We used the transformers library to do fine tuning regarding to our dataset.
- Evaluation:
    - Validation Set: Split the dataset into training, validation, and test sets to evaluate the model's performance.
    - Metrics: Calculate evaluation metrics using accuracy provided from the transformers

- Visualization: Visualize model predictions, and utilize a GUI using TKinter to display the predicted values and also ingredients.

IV. Experiments or tests that you have done.

Utilize the model and use multiple different unique new images from the internet to test the model with, to determine if the accuracy output from the scoring metric that we have selected is accurate. In this experimentation/test we tested 9 different pictures and the model got all of them correct

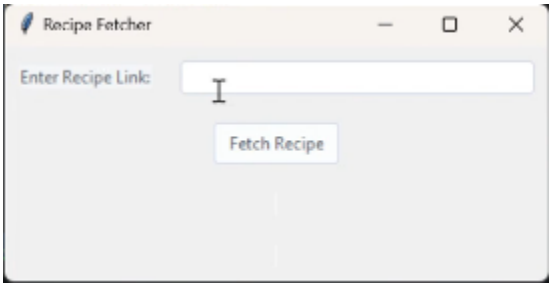




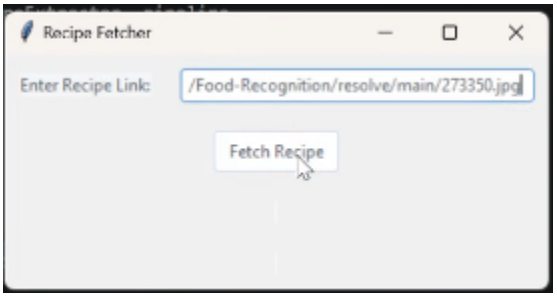
These images show the experiments our program have done

V. Program manual (with screenshots)

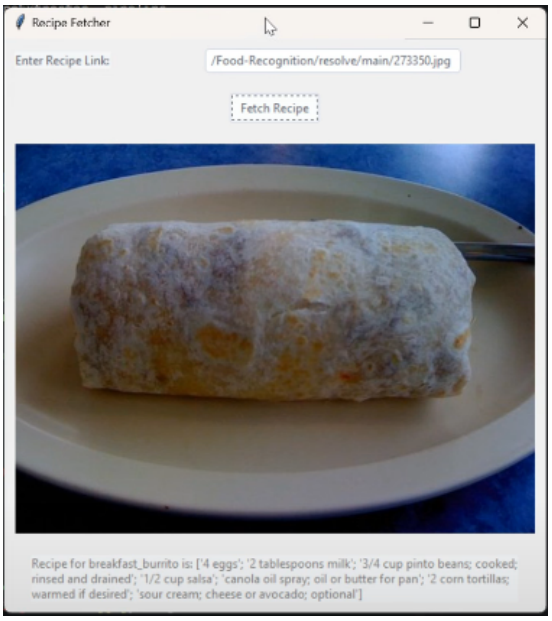
A. When the program is ran the user will be greeted by a simple prompt to enter a file path of the image they want to classify and fetch a possible recipe of



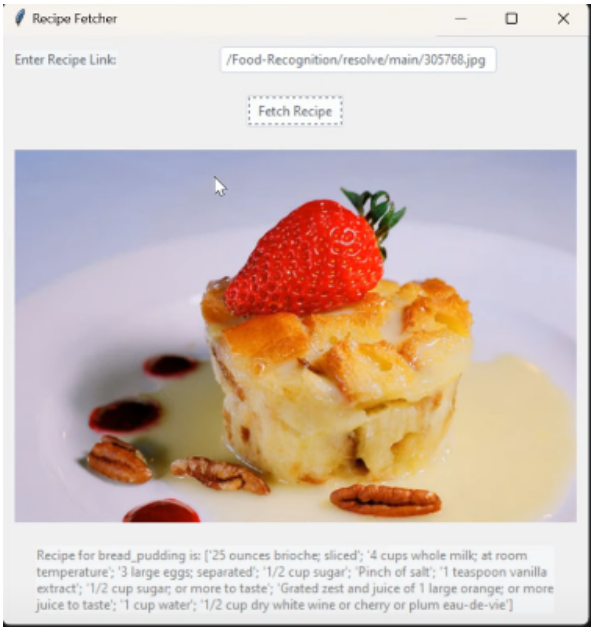
B. After entering the file recipe link/file path the users can click fetch recipe and obtain the classification/name of the food in the input



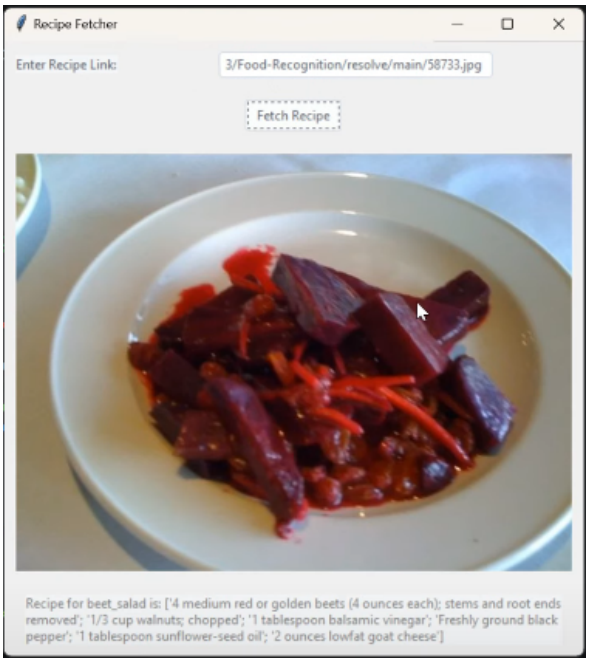
C. As you can see the program recognizes the image as a breakfast burrito and shows the recipe of the burrito



D. Here is an example using another different image



E. Another one



F. Essentially the user interface works by taking in a file path/link of the image that will then be passed through the model and outputs the recipe of the food.

VI. Link to video of the application demo (with max. length of 2 minutes)

https://youtu.be/lyvVBI8YPJM

VII. Link to your GIT repository

https://github.com/Jacques7103/AI_FP