FOOD INSIGHTS AND ANALYSIS BASED RECOMMENDATIONS

Project ID - 21_22-J-058

Project Proposal Report

T.Abishek

B.Sc. (Hons) in Information Technology Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

September 2021

FOOD INSIGHTS AND ANALYSIS BASED RECOMMENDATIONS

Project ID - 21_22-J-058

Project Proposal Report

T.Abishek

IT18234930

Supervising by

Ms.Anjali Gamage

B.Sc. (Hons) in Information Technology Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

September 2021

Declaration

We declare that this is our own work and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor Date

....

Abstract

People nowadays enjoy food photography because they enjoy food. Each meal has a backstory that is detailed in a complex recipe; unfortunately, we cannot learn about the preparation procedure by simply looking at a food image. As a result, we present a system that uses food photographs to reproduce cooking recipes. Our system uses a unique architecture to anticipate ingredients as sets, modeling their relationships without prescribing any order, and then creates cooking directions by simultaneously attending to both the image and the inferred components.

To create a recipe from an image, you must first comprehend the elements that make up the food, as well as any processing they underwent, such as slicing or mixing with other ingredients. The picture-to-recipe problem has traditionally been described as a retrieval challenge, in which a recipe is recovered from a fixed dataset using an embedding space image similarity score. The amount and diversity of the dataset, as well as the quality of the learnt embedding, have a significant impact on the performance of such systems. When a matching recipe for the picture query does not exist in the static dataset, these systems fail.

The image-to-recipe problem can be reformulated as a conditional generation problem to get around retrieval systems' dataset limits. We argue that a recipe-generation pipeline would benefit from an intermediate step: predicting the ingredients list, rather than getting the recipe directly from an image. The sequence of instructions would then be constructed based on both the image and its associated ingredient list, with the interaction between the two potentially providing further information about how the latter were processed to produce the final dish.

Our image-to-recipe technology takes a food image as input and produces a recipe with a title, ingredients, and cooking directions. Our solution begins with pretraining an image encoder and an ingredients decoder, which uses visual features extracted from the input image and ingredient co-occurrences to predict a set of ingredients. Then we train the ingredient encoder and instruction decoder, which create title and instructions by feeding

visual cues from the image and expected ingredients into a state-of-the-art sequence creation model.

Table of Contents

List of figures

List of tables

1 Introduction

1.1 Background & Literature survey

Food is necessary for human survival. It not only gives us energy, but it also defines our identity and culture [10]. We are what we eat, as the old adage goes, and food-related activities like cooking, eating, and talking about it occupy a big portion of our daily lives. Food culture has spread faster than ever in the digital age, with many people posting photos of food they're consuming on social media [11]. Searching for #food on Instagram yields at least 300 million results, while searching for #foodie yields at least 100 million results, demonstrating the undeniable value of food in our society. Furthermore, eating habits and cooking culture have changed over time. Food was traditionally produced at home, however nowadays we regularly consume food prepared by others (e.g. takeaways, catering and restaurants). As a result, access to specific information regarding prepared foods is limited, making it difficult to determine exactly what we eat. As a result, we believe that inverse cooking systems, which can infer components and cooking directions from a prepared meal, are required.

Visual recognition tasks such as natural image classification, object detection, and semantic segmentation have seen significant advancements in recent years [12]. Food identification, however, presents extra hurdles as compared to natural picture interpretation, because food and its components have substantial intraclass variability and severe deformations that occur throughout the cooking process. In a cooked food, ingredients are frequently hidden and come in a variety of colors, shapes, and textures. Visual ingredient detection also necessitates advanced reasoning and prior knowledge (e.g. cake will likely contain sugar and not salt, while croissant will presumably include butter). As a result, food recognition forces current computer vision systems to think beyond the obvious and utilize prior knowledge in order to provide high-quality structured food preparation descriptions.

Previous efforts to better comprehend food have primarily focused on categorizing foods and ingredients. A system for comprehensive visual food recognition, on the other hand, should be able to recognize not only the type of meal or its ingredients, but also the process of preparation. The picture-to-recipe problem has traditionally been framed as a retrieval job [3], in which a recipe is recovered from a fixed dataset using an embedding space image similarity score. The amount and diversity of the dataset, as well as the quality of the learnt embedding, have a significant impact on the performance of such systems. When a matching recipe for the picture query does not exist in the static dataset, these systems fail.

The image-to-recipe problem can be reformulated as a conditional generation problem to get around retrieval systems' dataset limits. As a result, we propose a system in this work that creates a cooking recipe from an image, complete with a title, ingredients, and cooking directions. Which generates cooking directions by first predicting ingredients from an image and then applying conditions to both the image and the ingredients. Our method is the first to produce culinary recipes directly from food photos, as far as we know. The instruction generation problem is modeled as a sequence generation problem with two modalities, namely an image and its expected constituents. We use their underlying structure to express the ingredient prediction problem as a set prediction. We represent ingredient dependencies without penalizing prediction order, reopening the debate over whether or not order matters. We tested our approach extensively on the large-scale Recipe1M dataset [13], which contains photos, ingredients, and cooking directions, and found it to be satisfactory. More specifically, we show that our inverse cooking system surpasses previously introduced image-to-recipe retrieval systems by a considerable margin in a human evaluation study. Furthermore, we demonstrate that food image-to-ingredient prediction is a difficult problem for humans, and that our approach is capable of outperforming them using a small amount of images.

1.2 Research Gap

Large-scale food datasets, such as Food-101 and Recipe1M [13], have enabled considerable gains in visual food recognition by providing reference benchmarks for training and comparing machine learning algorithms. As a result, there is already a large body of work in computer vision dealing with a wide range of food-related problems, with a particular emphasis on image classification [6], [14]. Following studies tackle more difficult tasks including estimating the number of calories in a given food image, estimating food quantities, predicting the list of present ingredients [3, 4], and determining the recipe for a given image [13]. [10] also gives an extensive cross-region analysis of culinary recipes, taking into account visuals, features (such as style and course), and recipe ingredients. Recipe generation has been studied in the context of creating procedural text from either flow graphs or ingredients' checklists in the natural language processing literature [15].

Deep neural networks for multi-label classification have received a lot of attention in the literature, with models [16] and loss functions that are ideally adapted for this job. Early approaches relied on single-label classification models with binary logistic loss [3], assuming label independence and discarding potentially relevant data. Label powersets are one method of capturing label dependencies. For large-scale issues, powersets evaluate all possible label combinations, making them unsolvable. Another expensive option is to calculate the labels' combined probability. To address this problem, probabilistic classifier chains and their recurrent neural network-based [16] counterparts propose decomposing the joint distribution into conditionals at the cost of intrinsic ordering. It's worth noting that most of these models ask you to forecast each of the possible labels. In addition, to preserve correlations and predict label sets, joint input and label embeddings have been established. Researchers have attempted to forecast the cardinality of a set of labels as an alternative, assuming label independence. Binary logistic loss [3], target distribution crossentropy [17], target distribution mean squared error, and ranking-based losses have all been explored and contrasted when it comes to multi-label classification objectives. The potential of the target distribution loss has been highlighted by recent results on large scale datasets [17].

In the literature, both text-based and image-based conditionings have been used to study conditional text generation with auto-regressive models. Different architecture designs, such as recurrent neural networks, convolutional models, and attention-based techniques, have been researched in neural machine translation, where the goal is to anticipate the translation for a given source text into another language. Sequenceto-sequence models have lately been used in more open-ended generation tasks like poetry and tale generation. Autoregressive models have shown promising performance in picture captioning, where the goal is to provide a brief explanation of the image contents, opening the door to less limited issues like creating descriptive paragraphs [18] or visual storytelling.

1.3 Research Problem

Creating a recipe (title, ingredients, and instructions) from a photograph is a difficult task that necessitates a comprehension of both the ingredients that make up the dish and the transformations they underwent, such as slicing, blending, or mixing with other ingredients. We claim that rather than extracting the recipe directly from an image, a recipe creation pipeline would benefit from an intermediate phase that predicts the ingredients list. The sequence of instructions would then be constructed based on both the image and its associated ingredient list, with the interaction between the two potentially providing further information about how the latter were processed to produce the final dish.

Our recipe creation system takes a food image as an input and generates a series of cooking instructions using an instruction decoder that takes two embeddings as input. The first encodes the elements derived from the image, while the second reflects visual features extracted from the image. To begin, we'll describe our transfomer-based instruction decoder. This allows us to conduct a formal assessment of the transformer, which we will then examine and tweak in order to anticipate constituents in a more orderly manner.

With the purpose of determining whether ingredients should be considered as lists or sets, we compare the proposed ingredient prediction algorithms to previously introduced models. As a starting point, we use models from the multilabel classification literature and tweak them to fit our needs. On the one hand, we have models that are trained to predict sets of ingredients using feed forward convolutional networks. Sequential models, on the other hand, predict lists by imposing order and exploiting interdependencies between components. Finally, we look at newly proposed models that combine set prediction and cardinality prediction to choose which components to include in the set [19].

2 Objectives

2.1 Main Objectives

The main goal of this research component is to recognize the image and predict the ingredients from a food image, as well as provide information on how the food was prepared and the name of the dish. The image processing technique makes it simple to detect food items. The food image will be captured when the system captures or inputs an image. Instead of getting the recipe directly from an image, an intermediate process anticipating the ingredients list is used. The sequence of instructions would then be constructed based on both the image and the list of materials associated with it, with the interaction between the two potentially providing additional insights. The system will next prepare the processed to make the final dish using preprocessing techniques and appropriate algorithms. As a result, this system will improve upon existing studies in generating recipes from image processing.

2.2 Specific Objectives

According to our research we identify the specific objectives that needed to include in our system In order to reach the main objectives, the specific objectives that needs to be attained is as follows,

- Image processing
- Analyze and identifying suitable recipe.
- Handling and getting exact dataset
- Identifying more suitable algorithm for each component
- Develop android application which should be user friendly.
- Train the dataset

3 Methodology

According to human assessment, our system proposes a new technique to generate recipes directly from food photos that delivers more engaging recipes than retrieval-based approaches. This technique increases ingredient prediction performance over previous baselines when tested on the large-scale Recipe1M dataset. We hope that by submitting a food photograph, we will be able to enable access to meal preparation.

Our image-to-recipe technology takes a food image as input and produces a recipe with a title, ingredients, and cooking directions. Our solution begins with pretraining an image encoder and an ingredients decoder, which uses visual features extracted from the input image and ingredient co-occurrences to predict a set of ingredients. Then we train the ingredient encoder and instruction decoder, which create title and instructions by feeding visual cues from the image and expected ingredients into a state-of-the-art sequence creation model.

Food recognition pushes today's computer vision systems to think beyond the obvious. Visual ingredient prediction necessitates higher-level thinking and prior knowledge when compared to natural image understanding (e.g., that croissants likely contain butter). Food components have substantial intra-class variability, heavy deformations occur during cooking, and elements are commonly occluded in a cooked dish, posing further complications. Our method is a first step toward a more comprehensive understanding of food.

The result of our research will be an Android application with three key capabilities or subsystems. Image processing and other machine learning, optimization, and visualization algorithms were used to create my component.

3.1 System Architecture

The overall group system architecture is shown in the Figure 1.

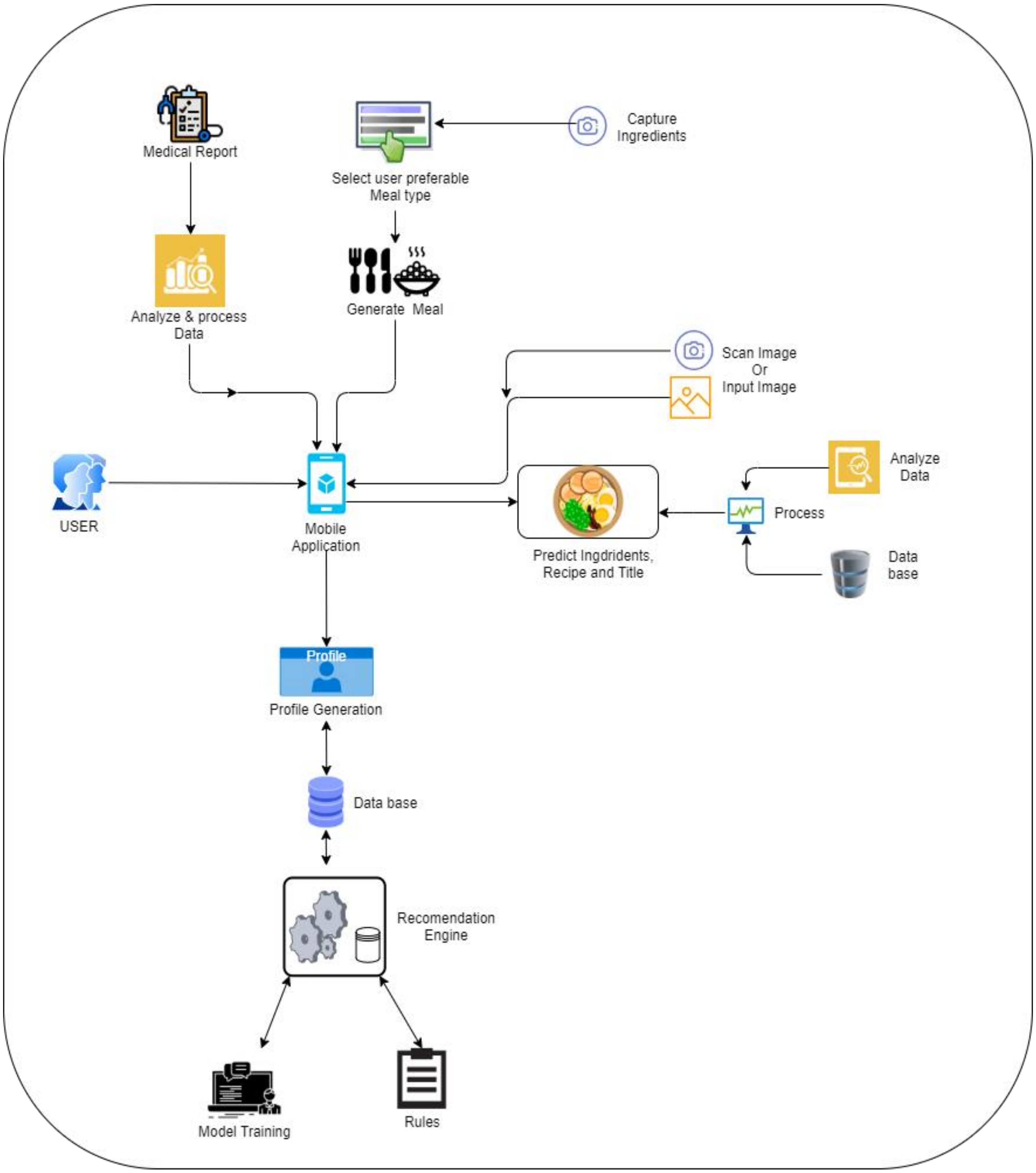


Figure 1-High Level Architectural Diagram

3.2 Work Breakdown Structure

The work breakdown Structure is shown in the Figure 2.

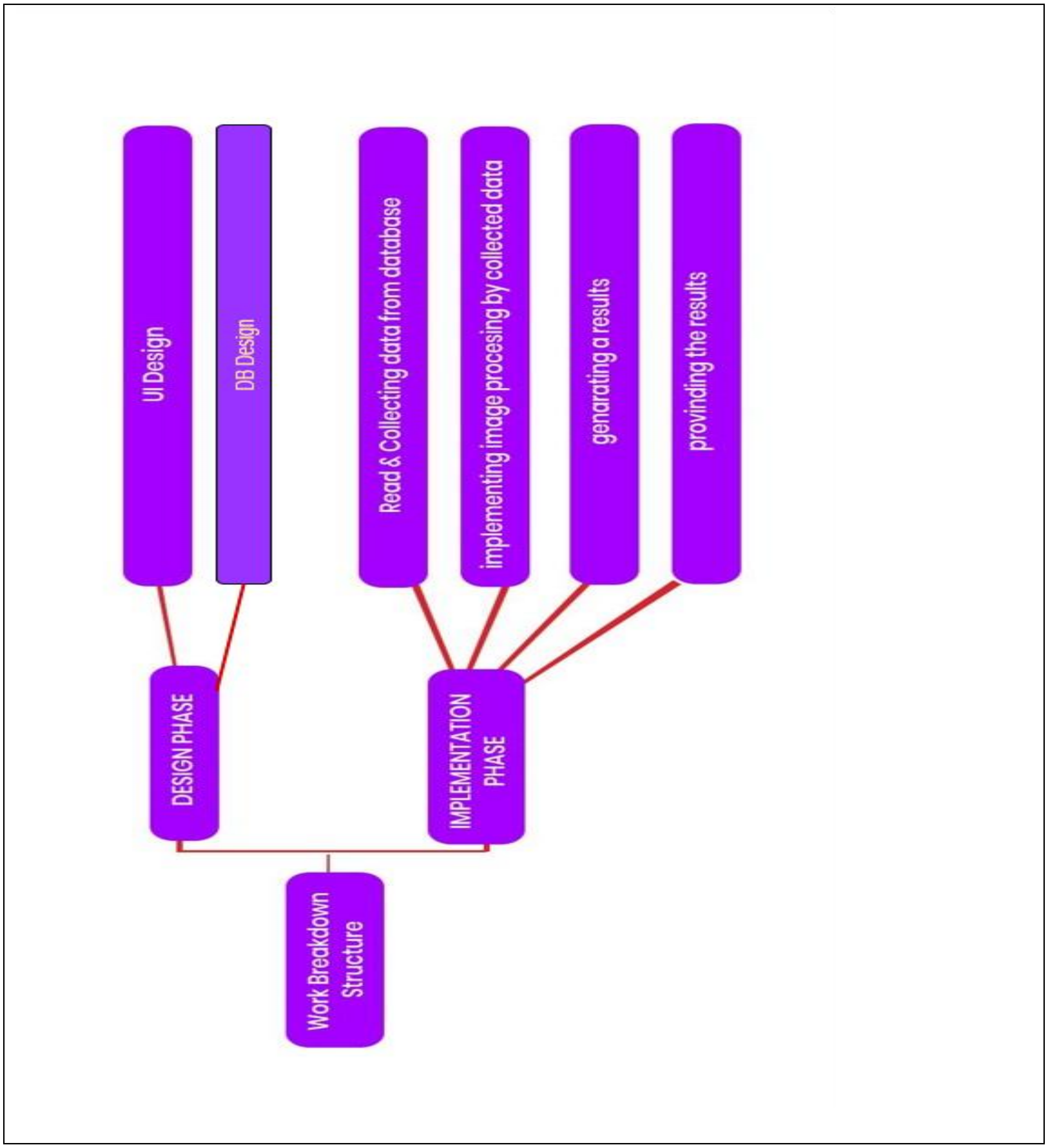


Figure 2- Work Breakdown Structure

3.3 Software Solution

The software development life cycle which would be considered will be the agile methodology. And Scrum will be the methodology that will be followed under the agile methodology. Scrum is a lightweight agile project management framework with broad applicability for managing and controlling iterative and incremental projects of all types [9]. Since scrum has the capability of inspecting and adapting to the change in requirements (Figure - 2), thus the solution the authors will implement is based on the hypothesis that was done by the literature survey and the survey implemented, constant changes

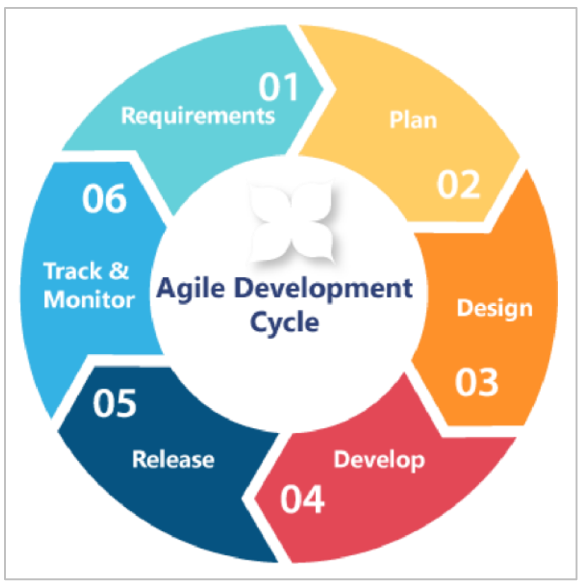


Figure 3- Agile methodology

3.4 Requirements

Technologies, techniques to be used

- React Native
- Python
- Firebase
- Image Processing
- DNN
- ML
- Tensor Flow
- NLP
- Dataset - http://im2recipe.csail.mit.edu/dataset/download/

Functional Requirements

- Image processing on identifying food ingredients and recipe
- According to collected data analyze and train them.
- Generate output from analyzed data.

Non-functional Requirements

- Security
- High Accuracy
- Availability
- Usability
- Performance
- Reliability

4. Description of Personal and Facilities

The overall group Description of Personal and Facilities is shown in Table 1.

Student ID	**Name**	**Work Load**
IT17111652	A.Prakash	Recommending recipe based on user preferences. • Create user profile to represent users having different tastes with text inputs. • Create a DNN model with different recipes. • Train the model along with the user profile and recipes. • Predict the best recipe for the users preference • Implement a food history tracker to prevent recommending the same recipe continuously.
IT18234930	T.Abishek	Identify the Food and predict the ingredients and how it was cooked. • Identify the input Image

		• Predict the tittle for the Food • Predict the ingredients from the input image • Predict the cooking instruction (recipe) for the image.
IT18228618	J. Abishek	Recommend recipe based on ingredients for chronically ill person. • Capture ingredients by image processing • Identify and segment ingredients • Analyze user medical condition • Recommend meal plans for user according to their health condition • Guide user pros and cons about what they have picked

Table 1- Description of Personal and Facilities

5 Gantt chart

The Gantt chart of the development process is as depicted in Figure 3.

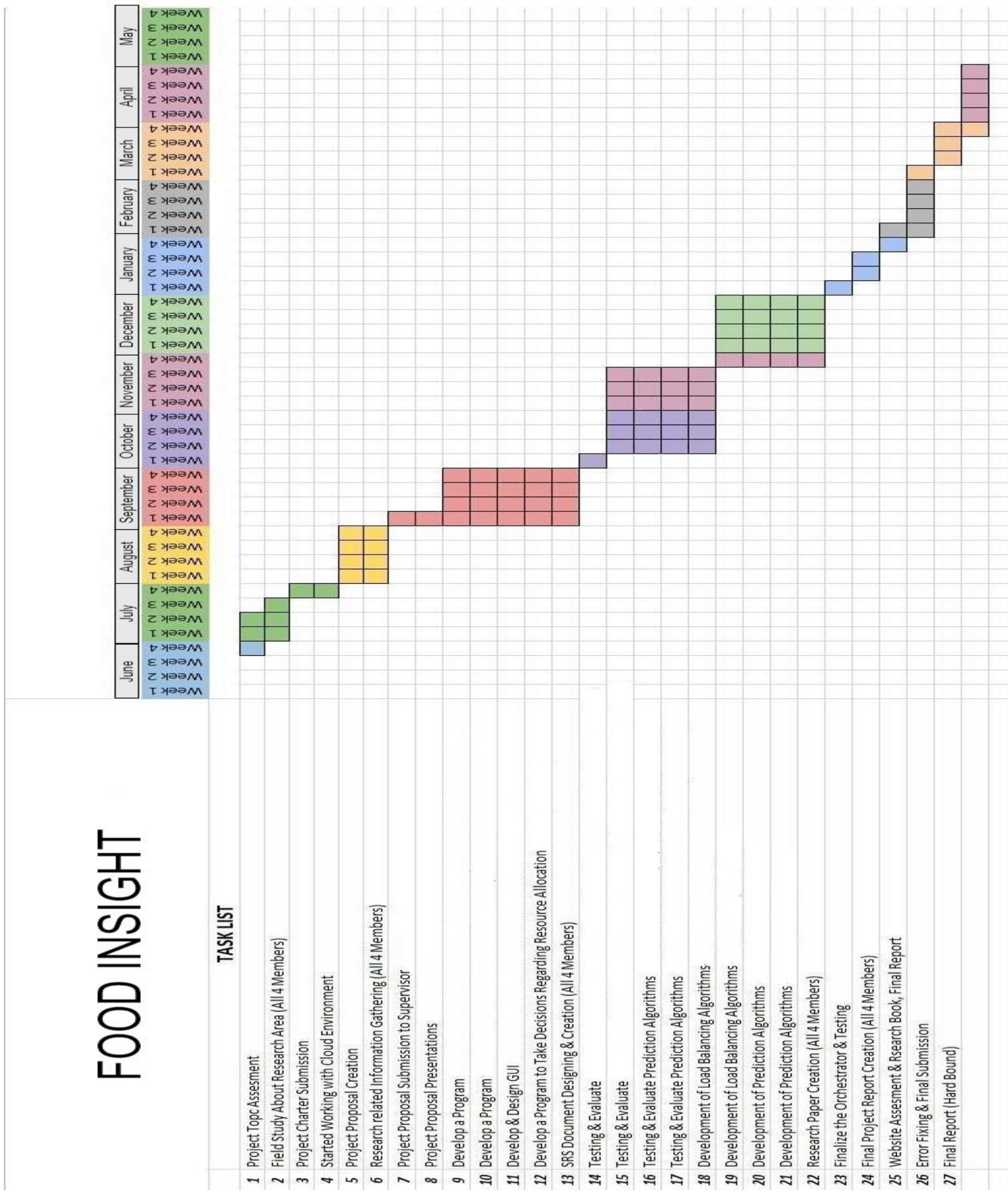
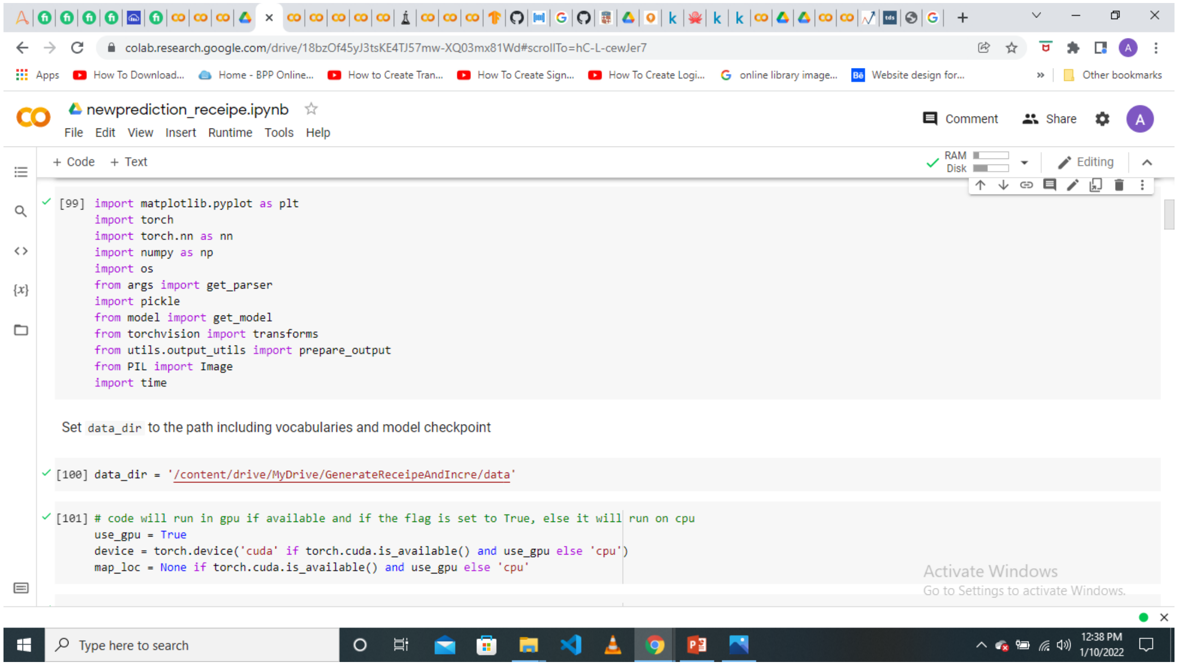
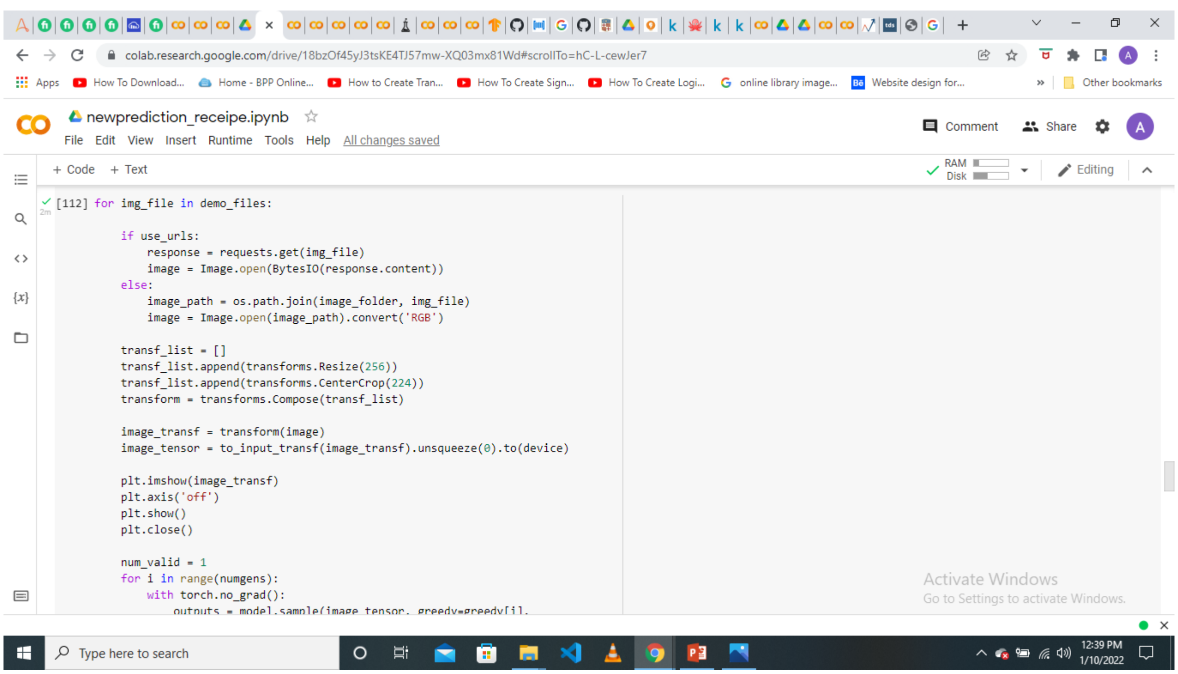
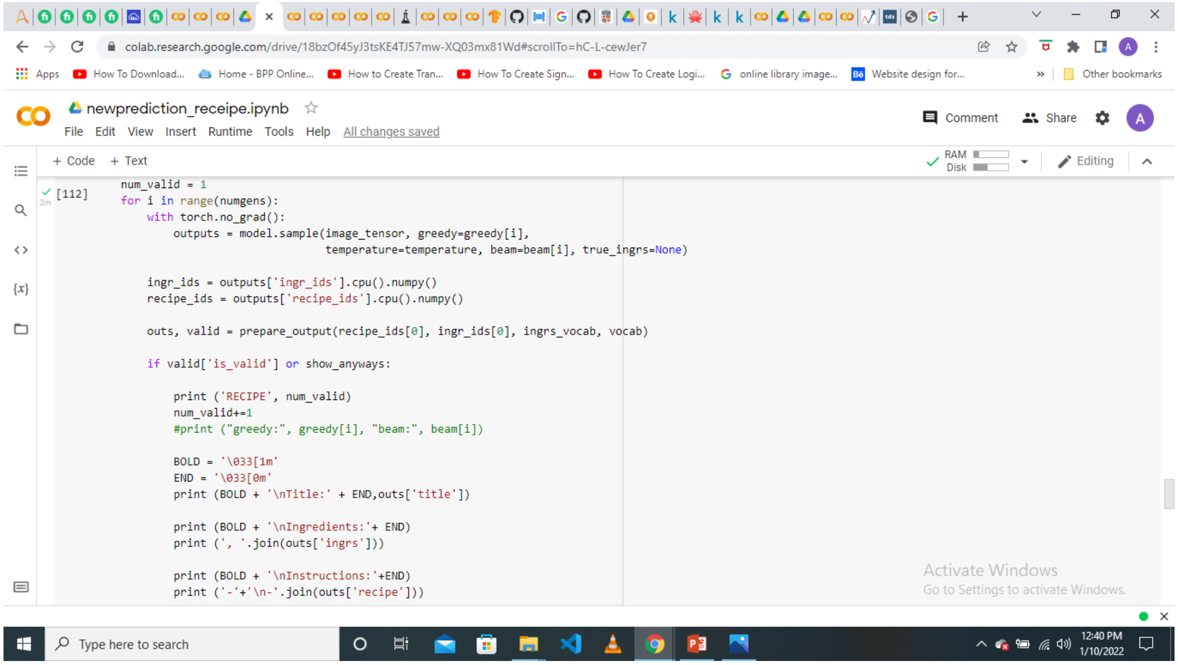


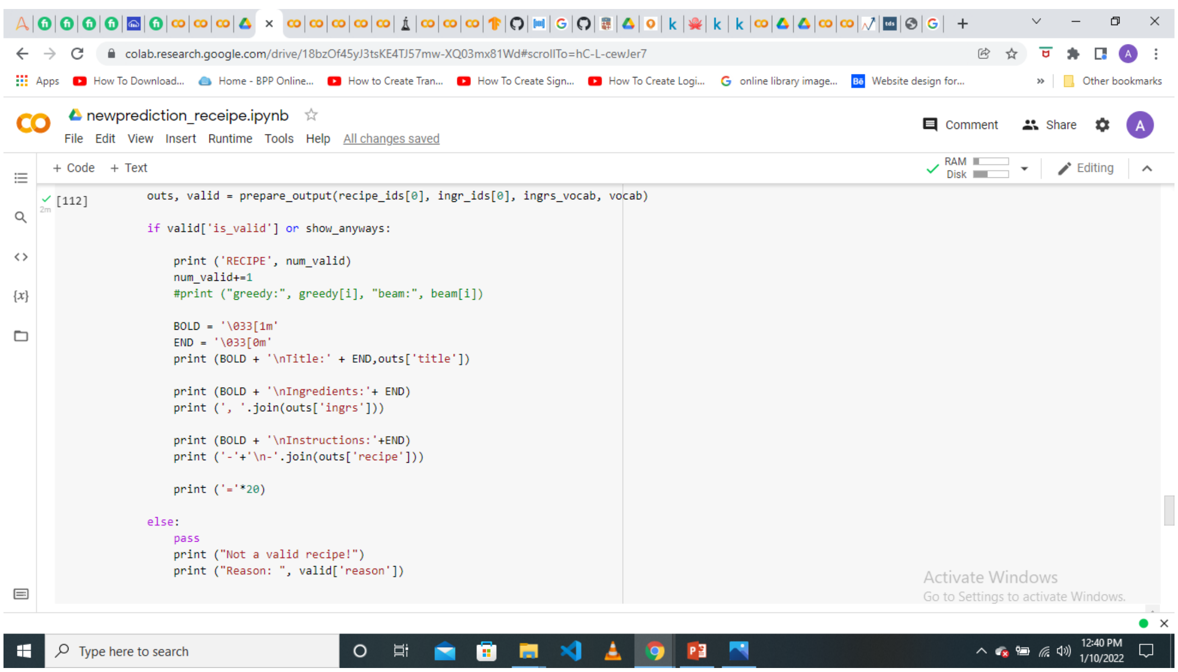
Figure 4- Gantt chart

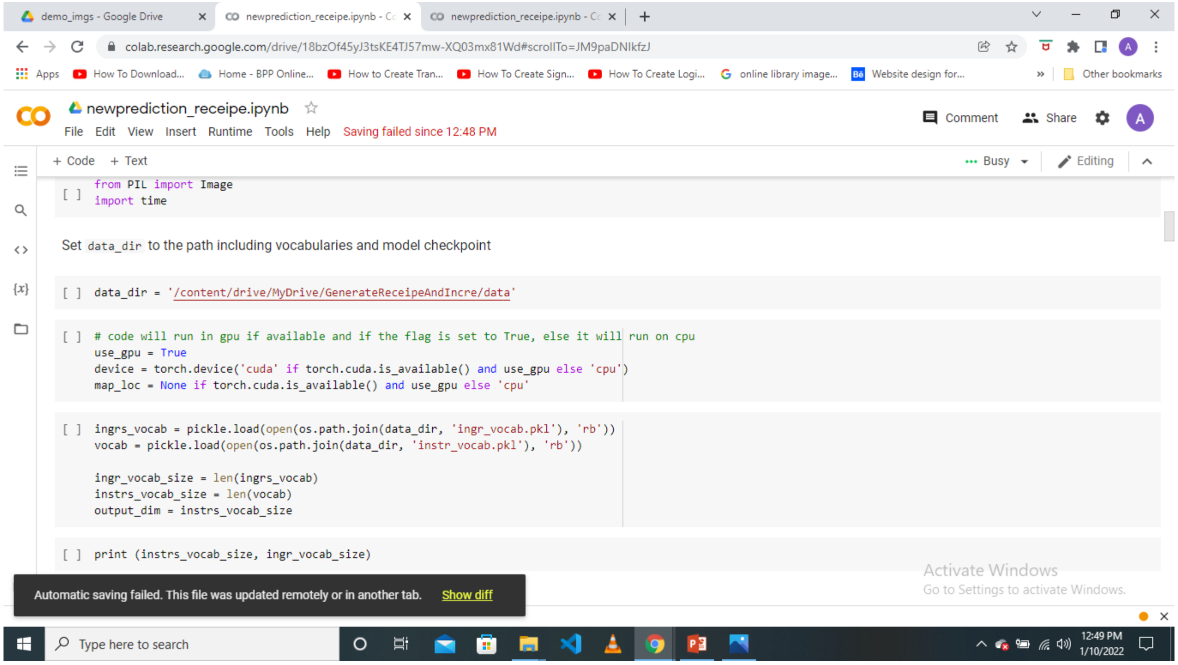
6 Result

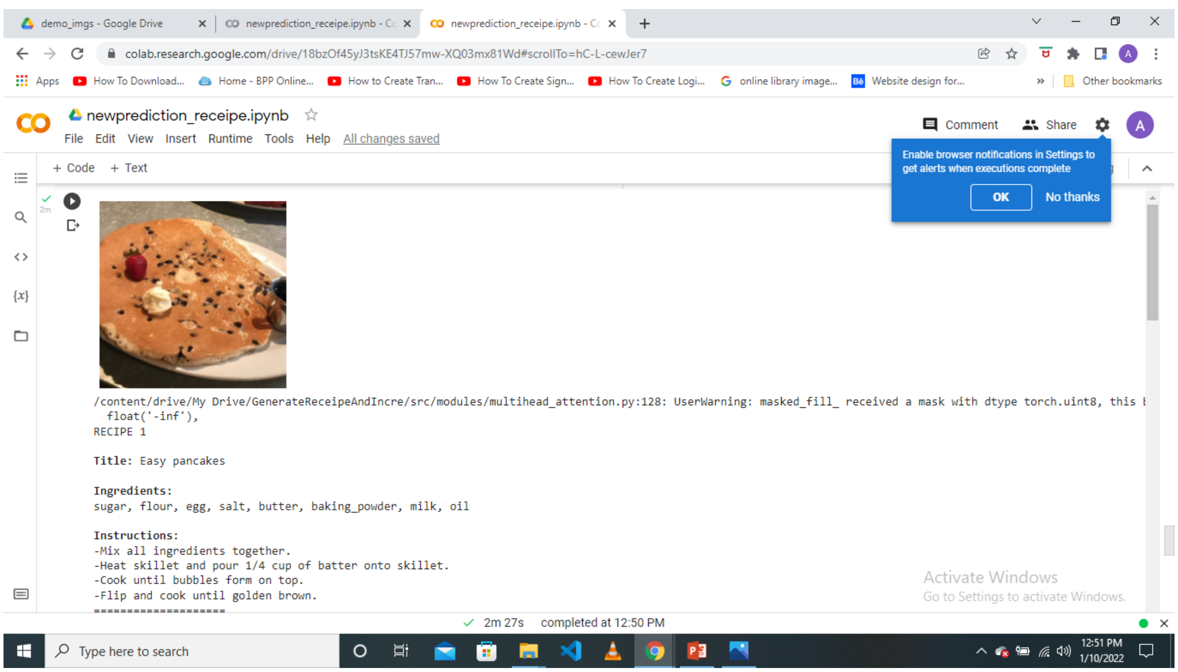

newprediction_receipe.ipynb
[99] import matplotlib.pyplot as plt
import torch
import torch.nn as nn
import numpy as np
import os
from args import get_parser
import pickle
from model import get_model
from torchvision import transforms
from utils.output_utils import prepare_output
from PIL import Image
import time
Set data_dir to the path including vocabularies and model checkpoint
[100] data_dir = '/content/drive/MyDrive/GenerateReceipeAndIncre/data'
[101] # code will run in gpu if available and if the flag is set to True, else it will run on cpu
use_gpu = True
device = torch.device('cuda' if torch.cuda.is_available() and use_gpu else 'cpu')
map_loc = None if torch.cuda.is_available() and use_gpu else 'cpu'

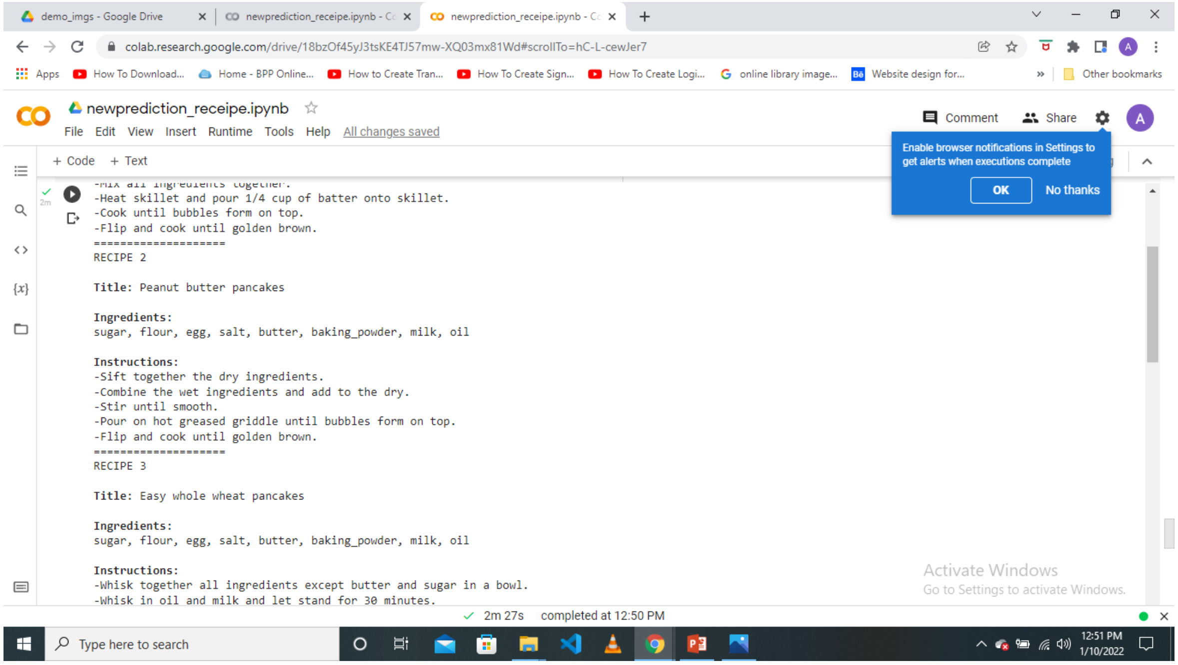

newprediction_receipe.ipynb
[112] for img_file in demo_files:
if use_urls:
response = requests.get(img_file)
image = Image.open(BytesIO(response.content))
else:
image_path = os.path.join(image_folder, img_file)
image = Image.open(image_path).convert('RGB')
transf_list = []
transf_list.append(transforms.Resize(256))
transf_list.append(transforms.CenterCrop(224))
transform = transforms.Compose(transf_list)
image_transf = transform(image)
image_tensor = to_input_transf(image_transf).unsqueeze(0).to(device)
plt.imshow(image_transf)
plt.axis('off')
plt.show()
plt.close()
num_valid = 1
for i in range(numgens):
with torch.no_grad():

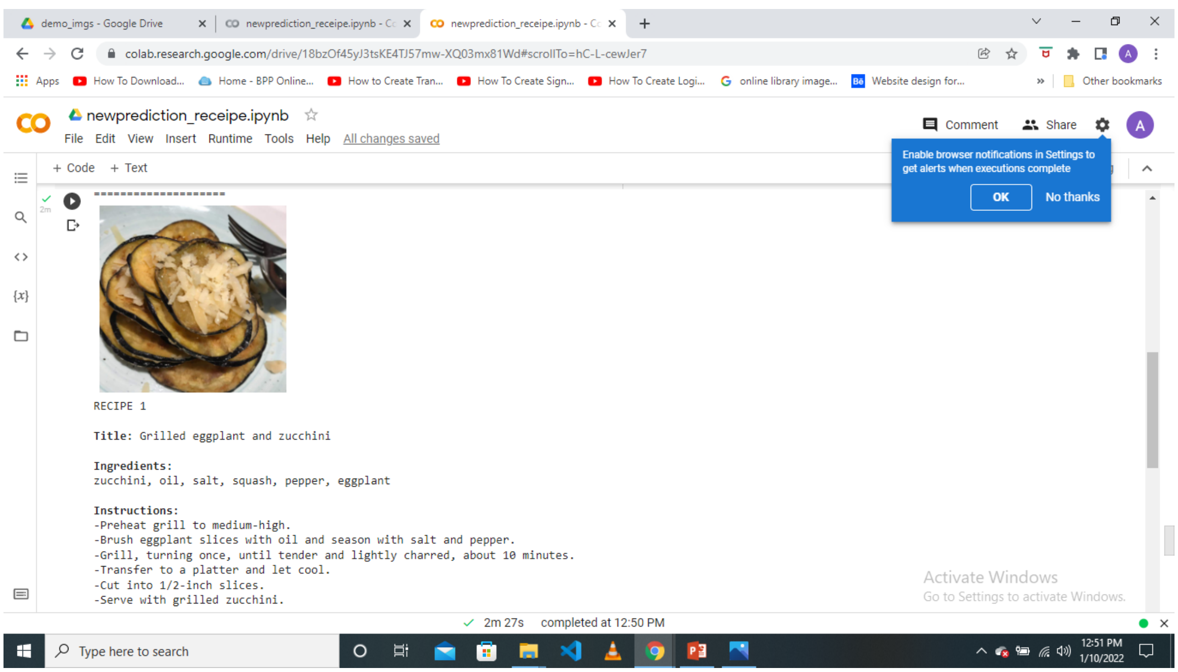

newprediction_receipe.ipynb
File Edit View Insert Runtime Tools Help All changes saved
[112]
num_valid = 1
for i in range(numgens):
    with torch.no_grad():
        outputs = model.sample(image_tensor, greedy=greedy[i],
                               temperature=temperature, beam=beam[i], true_ingrs=None)
    ingr_ids = outputs['ingr_ids'].cpu().numpy()
    recipe_ids = outputs['recipe_ids'].cpu().numpy()
    outs, valid = prepare_output(recipe_ids[0], ingr_ids[0], ingrs_vocab, vocab)
    if valid['is_valid'] or show_anyways:
        print ('RECIPE', num_valid)
        num_valid+=1
        #print ("greedy:", greedy[i], "beam:", beam[i])
        BOLD = '\033[1m'
        END = '\033[0m'
        print (BOLD + '\nTitle:' + END,outs['title'])
        print (BOLD + '\nIngredients:'+ END)
        print (', '.join(outs['ingrs']))
        print (BOLD + '\nInstructions:'+END)
        print ('-'+'\n-'.join(outs['recipe']))
Activate Windows
Go to Settings to activate Windows.
Type here to search
12:40 PM
1/10/2022

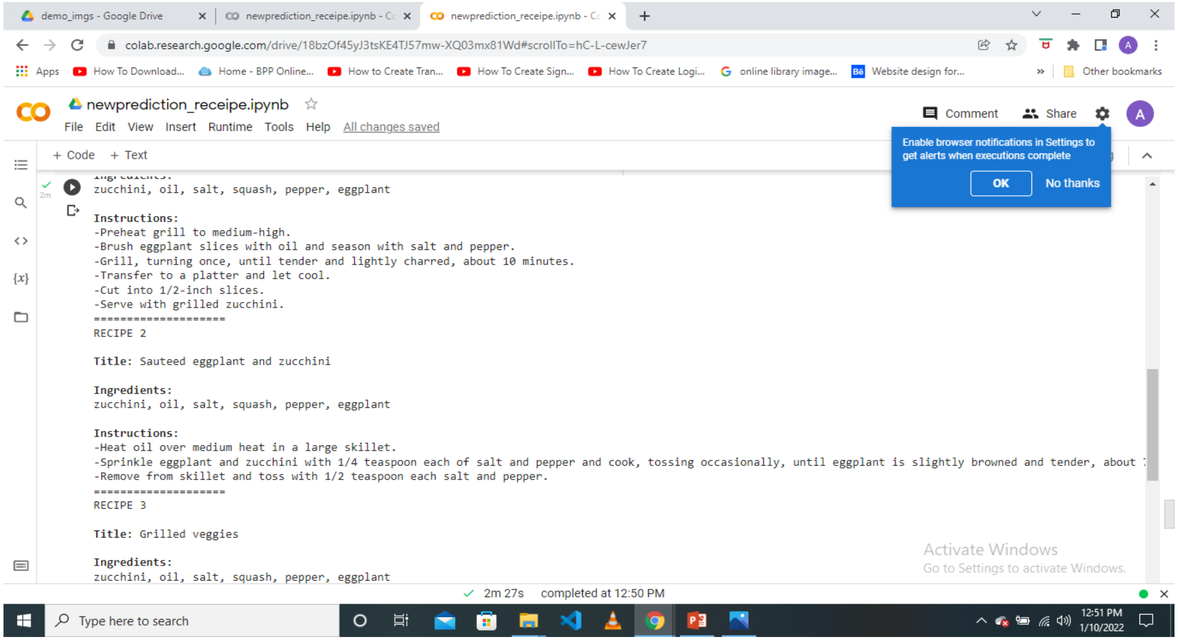

newprediction_receipe.ipynb
File Edit View Insert Runtime Tools Help All changes saved
[112]
    outs, valid = prepare_output(recipe_ids[0], ingr_ids[0], ingrs_vocab, vocab)
    if valid['is_valid'] or show_anyways:
        print ('RECIPE', num_valid)
        num_valid+=1
        #print ("greedy:", greedy[i], "beam:", beam[i])
        BOLD = '\033[1m'
        END = '\033[0m'
        print (BOLD + '\nTitle:' + END,outs['title'])
        print (BOLD + '\nIngredients:'+ END)
        print (', '.join(outs['ingrs']))
        print (BOLD + '\nInstructions:'+END)
        print ('-'+'\n-'.join(outs['recipe']))
        print ('='*20)
    else:
        pass
        print ("Not a valid recipe!")
        print ("Reason: ", valid['reason'])
Activate Windows
Go to Settings to activate Windows.
Type here to search
12:40 PM
1/10/2022

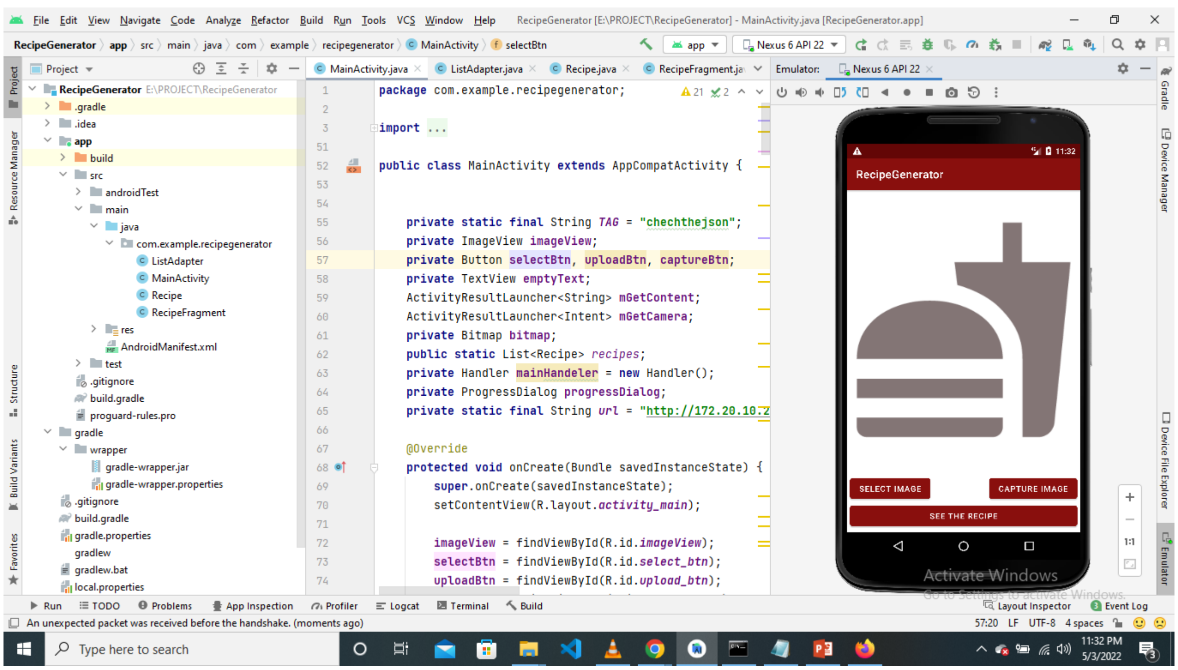

newprediction_receipe.ipynb
File Edit View Insert Runtime Tools Help Saving failed since 12:48 PM
+ Code + Text
from PIL import Image
import time
Set data_dir to the path including vocabularies and model checkpoint
data_dir = '/content/drive/MyDrive/GenerateReceipeAndIncre/data'
# code will run in gpu if available and if the flag is set to True, else it will run on cpu
use_gpu = True
device = torch.device('cuda' if torch.cuda.is_available() and use_gpu else 'cpu')
map_loc = None if torch.cuda.is_available() and use_gpu else 'cpu'
ingrs_vocab = pickle.load(open(os.path.join(data_dir, 'ingr_vocab.pkl'), 'rb'))
vocab = pickle.load(open(os.path.join(data_dir, 'instr_vocab.pkl'), 'rb'))
ingr_vocab_size = len(ingrs_vocab)
instrs_vocab_size = len(vocab)
output_dim = instrs_vocab_size
print (instrs_vocab_size, ingr_vocab_size)
Automatic saving failed. This file was updated remotely or in another tab. Show diff
Activate Windows
Go to Settings to activate Windows.
Type here to search
12:49 PM
1/10/2022

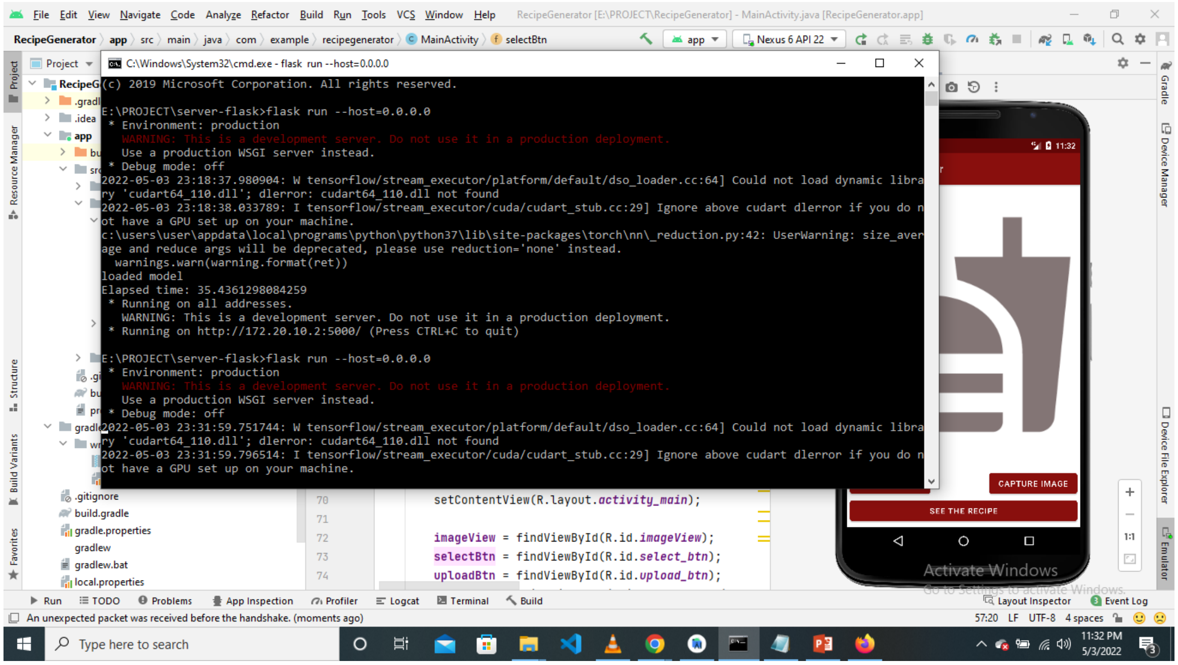

newprediction_receipe.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
Enable browser notifications in Settings to get alerts when executions complete
OK
No thanks
/content/drive/My Drive/GenerateReceipeAndIncre/src/modules/multihead_attention.py:128: UserWarning: masked_fill_ received a mask with dtype torch.uint8, this
float('-inf'),
RECIPE 1
Title: Easy pancakes
Ingredients:
sugar, flour, egg, salt, butter, baking_powder, milk, oil
Instructions:
-Mix all ingredients together.
-Heat skillet and pour 1/4 cup of batter onto skillet.
-Cook until bubbles form on top.
-Flip and cook until golden brown.
2m 27s completed at 12:50 PM
Activate Windows
Go to Settings to activate Windows.
Type here to search
12:51 PM
1/10/2022


newprediction_receipe.ipynb
-Heat skillet and pour 1/4 cup of batter onto skillet.
-Cook until bubbles form on top.
-Flip and cook until golden brown.
====================
RECIPE 2
Title: Peanut butter pancakes
Ingredients:
sugar, flour, egg, salt, butter, baking_powder, milk, oil
Instructions:
-Sift together the dry ingredients.
-Combine the wet ingredients and add to the dry.
-Stir until smooth.
-Pour on hot greased griddle until bubbles form on top.
-Flip and cook until golden brown.
====================
RECIPE 3
Title: Easy whole wheat pancakes
Ingredients:
sugar, flour, egg, salt, butter, baking_powder, milk, oil
Instructions:
-Whisk together all ingredients except butter and sugar in a bowl.
-Whisk in oil and milk and let stand for 30 minutes.
2m 27s completed at 12:50 PM

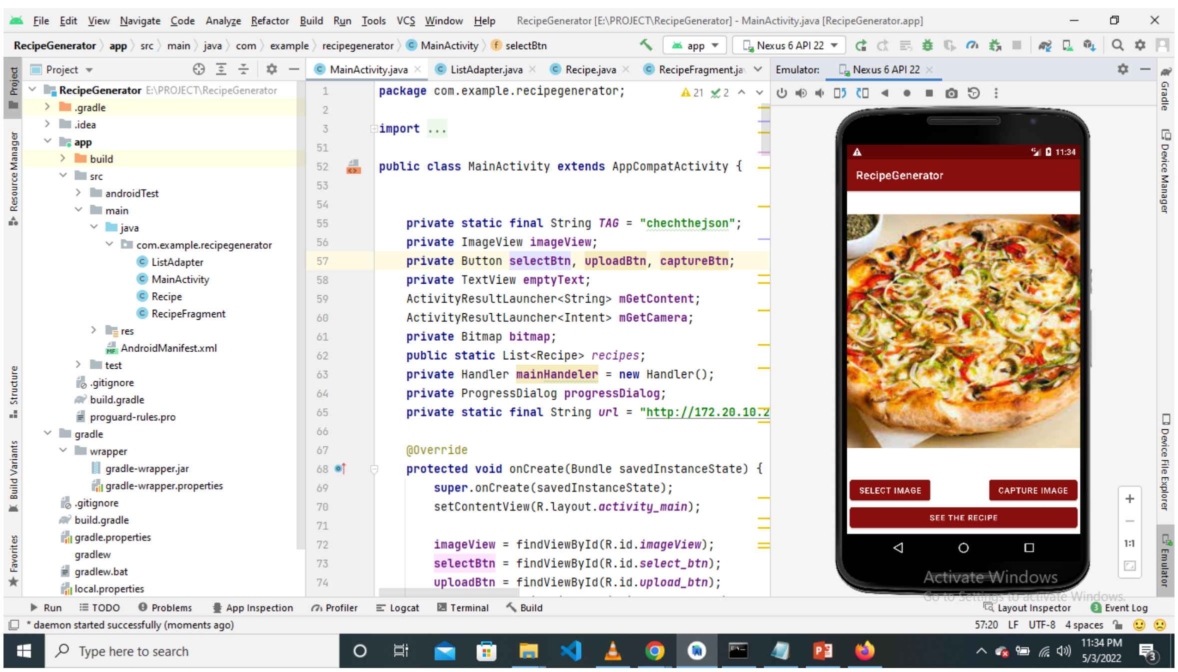

newprediction_receipe.ipynb
====================
RECIPE 1
Title: Grilled eggplant and zucchini
Ingredients:
zucchini, oil, salt, squash, pepper, eggplant
Instructions:
-Preheat grill to medium-high.
-Brush eggplant slices with oil and season with salt and pepper.
-Grill, turning once, until tender and lightly charred, about 10 minutes.
-Transfer to a platter and let cool.
-Cut into 1/2-inch slices.
-Serve with grilled zucchini.
2m 27s completed at 12:50 PM


newprediction_receipe.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
Comment Share
Enable browser notifications in Settings to get alerts when executions complete
OK No thanks
zucchini, oil, salt, squash, pepper, eggplant
Instructions:
-Preheat grill to medium-high.
-Brush eggplant slices with oil and season with salt and pepper.
-Grill, turning once, until tender and lightly charred, about 10 minutes.
-Transfer to a platter and let cool.
-Cut into 1/2-inch slices.
-Serve with grilled zucchini.
====================
RECIPE 2
Title: Sauteed eggplant and zucchini
Ingredients:
zucchini, oil, salt, squash, pepper, eggplant
Instructions:
-Heat oil over medium heat in a large skillet.
-Sprinkle eggplant and zucchini with 1/4 teaspoon each of salt and pepper and cook, tossing occasionally, until eggplant is slightly browned and tender, about
-Remove from skillet and toss with 1/2 teaspoon each salt and pepper.
====================
RECIPE 3
Title: Grilled veggies
Ingredients:
zucchini, oil, salt, squash, pepper, eggplant
2m 27s completed at 12:50 PM
Type here to search
12:51 PM
1/10/2022

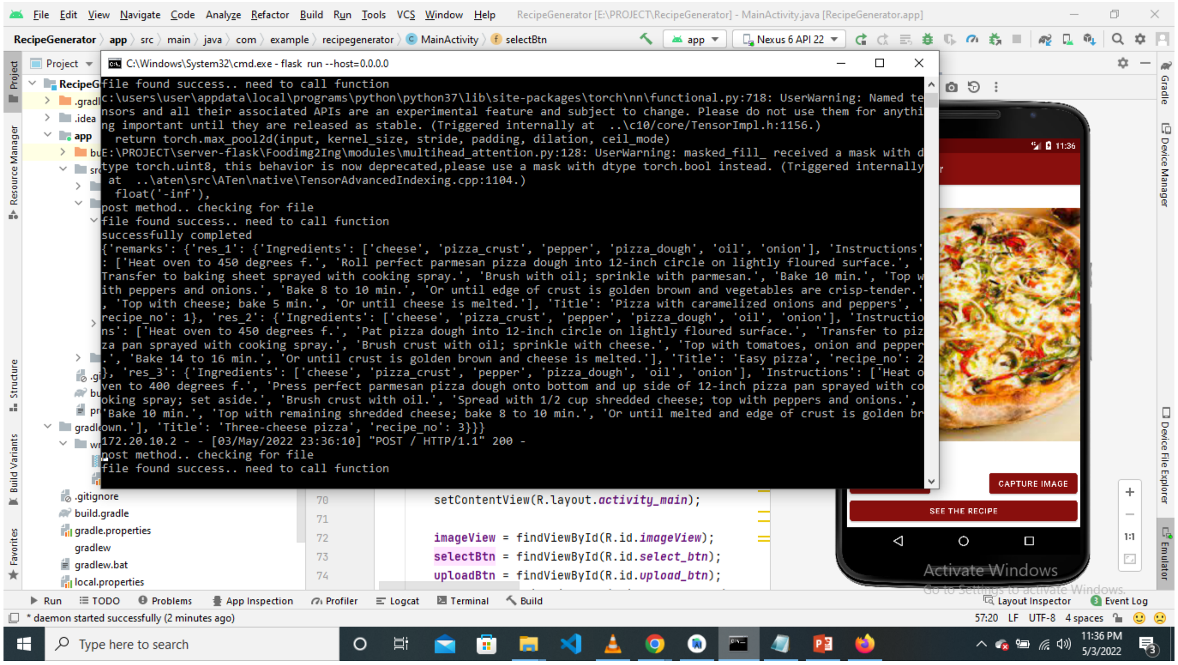

RecipeGenerator [E:\PROJECT\RecipeGenerator] - MainActivity.java [RecipeGenerator.app]
MainActivity.java ListAdapter.java Recipe.java RecipeFragment.java
package com.example.recipegenerator;
import ...
public class MainActivity extends AppCompatActivity {
private static final String TAG = "chechthejson";
private ImageView imageView;
private Button selectBtn, uploadBtn, captureBtn;
private TextView emptyText;
ActivityResultLauncher<String> mGetContent;
ActivityResultLauncher<Intent> mGetCamera;
private Bitmap bitmap;
public static List<Recipe> recipes;
private Handler mainHandeler = new Handler();
private ProgressDialog progressDialog;
private static final String url = "http://172.20.10.2
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
imageView = findViewById(R.id.imageView);
selectBtn = findViewById(R.id.select_btn);
uploadBtn = findViewById(R.id.upload_btn);
RecipeGenerator
SELECT IMAGE
CAPTURE IMAGE
SEE THE RECIPE
An unexpected packet was received before the handshake. (moments ago)
Type here to search
11:32 PM
5/3/2022

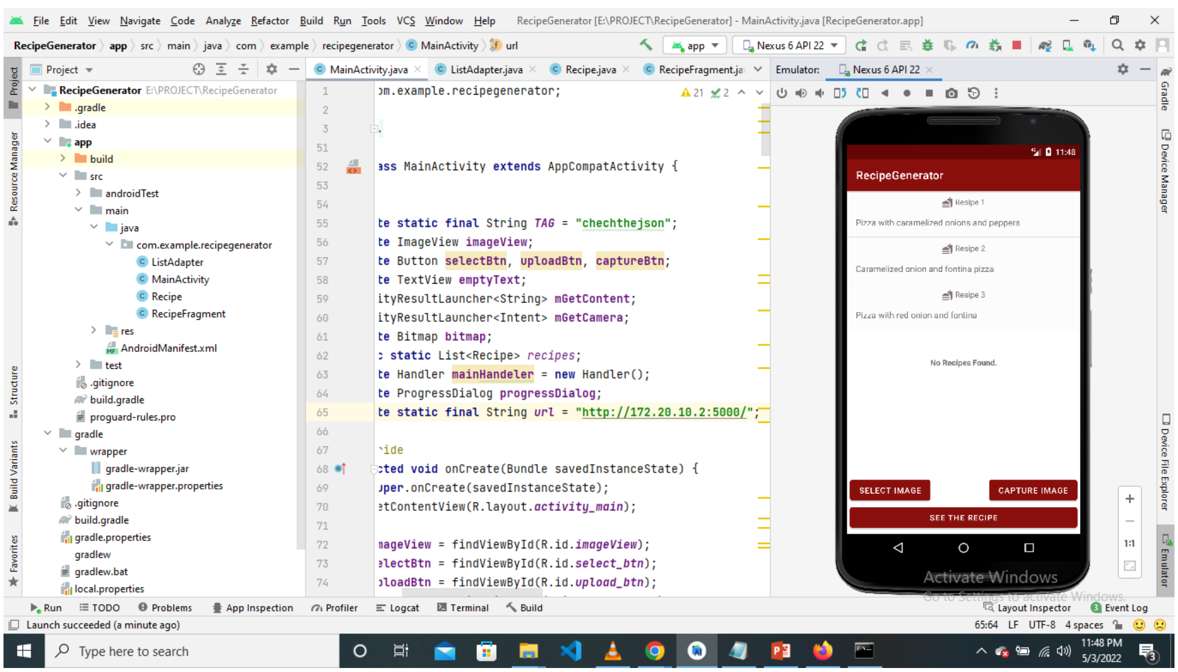


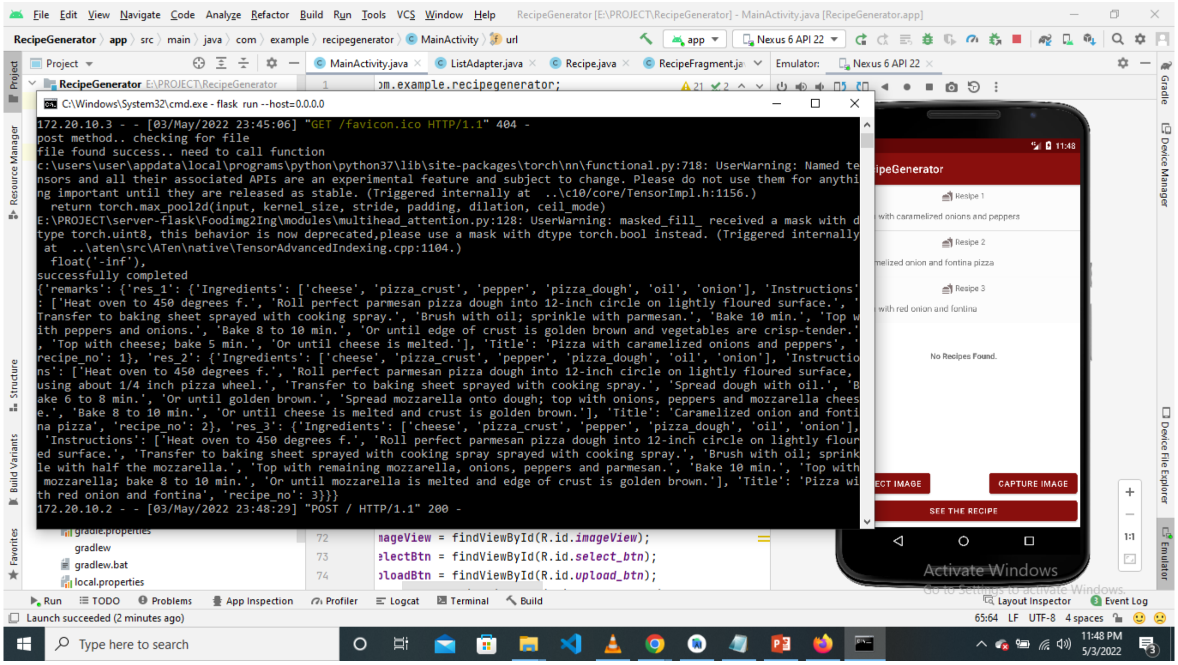
C:\Windows\System32\cmd.exe - flask run --host=0.0.0.0
(c) 2019 Microsoft Corporation. All rights reserved.
E:\PROJECT\server-flask>flask run --host=0.0.0.0
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
2022-05-03 23:18:37.980904: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dlerror: cudart64_110.dll not found
2022-05-03 23:18:38.033789: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
c:\users\user\appdata\local\programs\python\python37\lib\site-packages\torch\nn_reduction.py:42: UserWarning: size_average and reduce args will be deprecated, please use reduction='none' instead.
warnings.warn(warning.format(ret))
loaded model
Elapsed time: 35.4361298084259
* Running on all addresses.
WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://172.20.10.2:5000/ (Press CTRL+C to quit)
E:\PROJECT\server-flask>flask run --host=0.0.0.0
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
2022-05-03 23:31:59.751744: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dlerror: cudart64_110.dll not found
2022-05-03 23:31:59.796514: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
setContentView(R.layout.activity_main);
imageView = findViewById(R.id.imageView);
selectBtn = findViewById(R.id.select_btn);
uploadBtn = findViewById(R.id.upload_btn);
CAPTURE IMAGE
SEE THE RECIPE
An unexpected packet was received before the handshake. (moments ago)
Type here to search
11:32 PM
5/3/2022

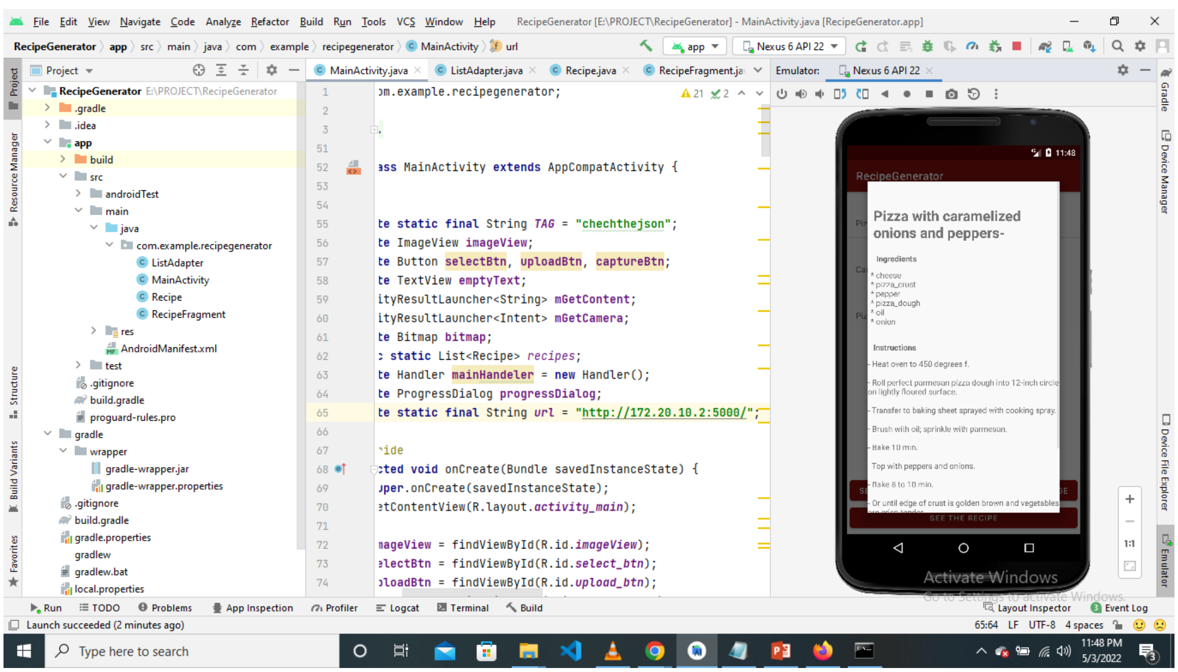


RecipeGenerator [E:\PROJECT\RecipeGenerator] - MainActivity.java [RecipeGenerator.app]
MainActivity.java
ListAdapter.java
Recipe.java
RecipeFragment.java
package com.example.recipegenerator;
import ...
public class MainActivity extends AppCompatActivity {
private static final String TAG = "chechthejson";
private ImageView imageView;
private Button selectBtn, uploadBtn, captureBtn;
private TextView emptyText;
ActivityResultLauncher<String> mGetContent;
ActivityResultLauncher<Intent> mGetCamera;
private Bitmap bitmap;
public static List<Recipe> recipes;
private Handler mainHandeler = new Handler();
private ProgressDialog progressDialog;
private static final String url = "http://172.20.10.2
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
imageView = findViewById(R.id.imageView);
selectBtn = findViewById(R.id.select_btn);
uploadBtn = findViewById(R.id.upload_btn);
RecipeGenerator
SELECT IMAGE
CAPTURE IMAGE
SEE THE RECIPE
daemon started successfully (moments ago)
Type here to search
11:34 PM
5/3/2022









7 Reference list

[1] Luis Herranz, Shuqiang Jiang, and Ruihan Xu. Modeling restaurant context for food recognition. IEEE Transactions on Multimedia, 2017.

[2] Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. Cleannet: Transfer learning for scalable image classifier training with label noise. In CVPR, 2018.

[3] Jing-Jing Chen and Chong-Wah Ngo. Deep-based ingredient recognition for cooking recipe retrieval. In ACM Multimedia. ACM, 2016.

[4] Niki Martinel, Gian Luca Foresti, and Christian Micheloni. Wide-slice residual networks for food recognition. In WACV, 2018.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2016.

[6] Xin Chen, Hua Zhou, and Liang Diao. Chinesefoodnet: A large-scale image dataset for chinese food recognition. CoRR, abs/1705.02743, 2017.

[7] Sara McGuire. Food Photo Frenzy: Inside the Instagram Craze and Travel Trend. https://www.business. com/articles/food-photo-frenzy-insidethe-instagram-craze-and-travel-trend/, 2017. [Online; accessed Nov-2018].

[8] https://www.researchgate.net/publication/329734580_Inverse_Cooking_Recipe_Generation_from_Food_Images

[9] VersionOne. (2018). Agile Methodologies for Software Development. [online] Available at: https://www.versionone.com/agile-101/agile-methodologies/ [Accessed 1 Apr. 2018].

[10] Weiqing Min, Bing-Kun Bao, Shuhuan Mei, Yaohui Zhu, Yong Rui, and Shuqiang Jiang. You are what you eat: Exploring rich recipe information for cross-region food analysis. IEEE Transactions on Multimedia, 2018.

[11] Sara McGuire. Food Photo Frenzy: Inside the Instagram Craze and Travel Trend. https://www.business. com/articles/food-photo-frenzy-insidethe-instagram-craze-and-travel-trend/, 2017. [Online; accessed Nov-2018].

[12] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.

[13] Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, and Antonio Torralba. Learning cross-modal embeddings for cooking recipes and food images. CVPR, 2017.

[14] Shota Horiguchi, Sosuke Amano, Makoto Ogawa, and Kiyoharu Aizawa. Personalized classifier for food image recognition. IEEE Transactions on Multimedia, 2018.

[15] Chloe Kiddon, Luke Zettlemoyer, and Yejin Choi. Globally ´ coherent text generation with neural checklist models. In EMNLP, 2016.

[16] Jinseok Nam, Eneldo Loza Menc´ıa, Hyunwoo J Kim, and Johannes Furnkranz. Maximizing subset accuracy with re- ¨ current neural networks in multi-label classification. In NeurIPS. 2017.

[17] Dhruv Mahajan, Ross B. Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. CoRR, abs/1805.00932, 2018.

[18] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In CVPR, 2017.

[19] S Hamid Rezatofighi, Anton Milan, Qinfeng Shi, Anthony Dick, and Ian Reid. Joint learning of set cardinality and state distribution. AAAI, 2018.

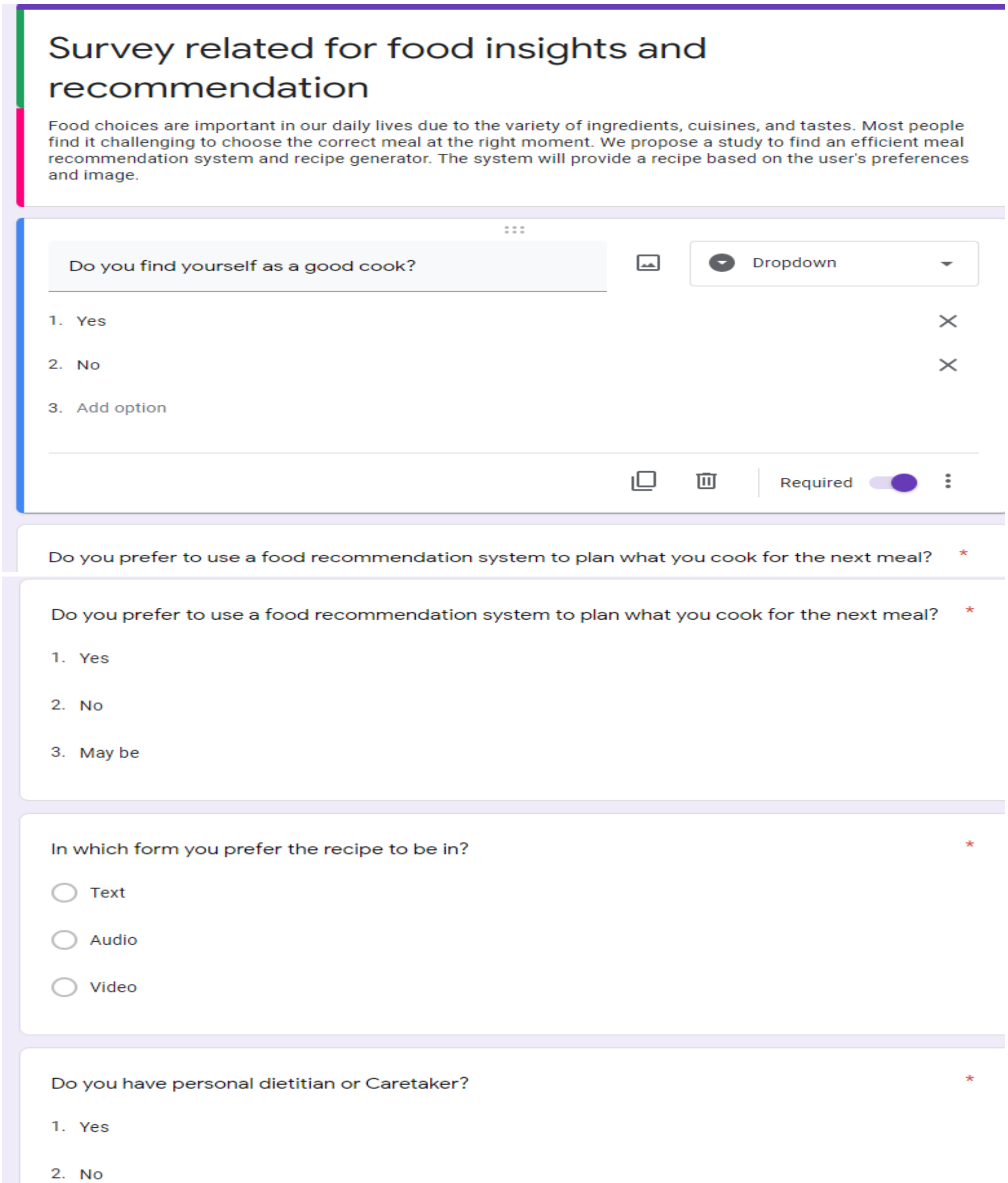
[20] D.-W. S. Nek Valous, "Image processing techniques for computer vision in the food and beverage industries," 2012.

[21] T. Mokdara, P. Pusawiro and J. Harnsomburana, "Personalized Food Recommendation Using Deep Neural Network," 2018.

[22] http://im2recipe.csail.mit.edu/dataset/download/

8 Appendices

8.1 Survey questions and response.



Survey related for food insights and recommendation

Food choices are important in our daily lives due to the variety of ingredients, cuisines, and tastes. Most people find it challenging to choose the correct meal at the right moment. We propose a study to find an efficient meal recommendation system and recipe generator. The system will provide a recipe based on the user's preferences and image.

Do you find yourself as a good cook?

Dropdown

1. Yes
2. No
3. Add option

Required

Do you prefer to use a food recommendation system to plan what you cook for the next meal? *

Do you prefer to use a food recommendation system to plan what you cook for the next meal? *

1. Yes
2. No
3. May be

In which form you prefer the recipe to be in? *

- Text
- Audio
- Video

Do you have personal dietitian or Caretaker? *

1. Yes
2. No

Do you have personal dietitian or Caretaker? *

1. Yes
2. No
3. Maybe

Are you aware of what food you should take for your diagnosis? *

1. Yes
2. No

You are a *

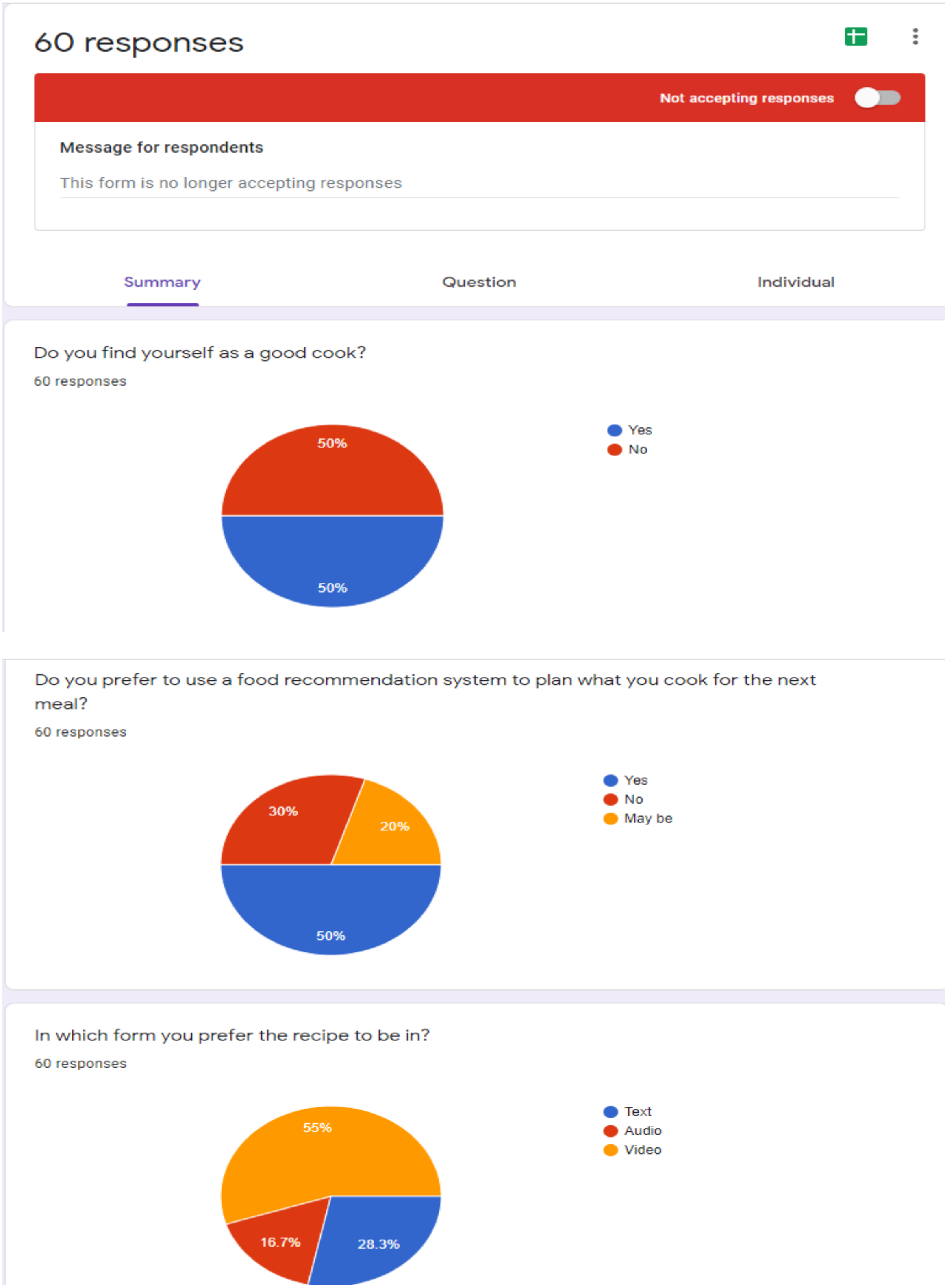
- [ ] Dietitian
- [ ] Nutritionist
- [ ] Athlete

You are a *

- [ ] Dietitian
- [ ] Nutritionist
- [ ] Athlete
- [ ] Common

Do you encourage this application to maintain your personal diet *

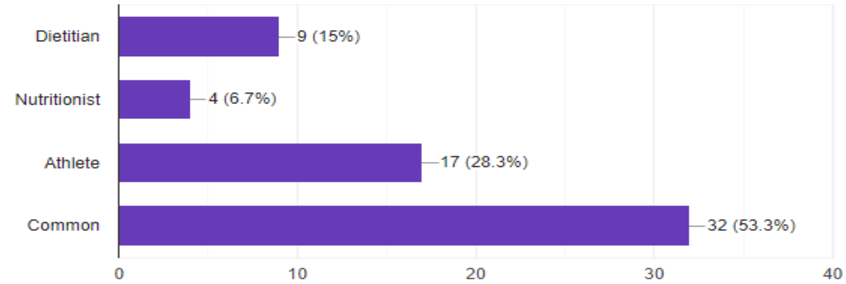
1. Yes
2. No
3. Maybe



60 responses
Not accepting responses
Message for respondents
This form is no longer accepting responses
Summary
Question
Individual
Do you find yourself as a good cook?
60 responses
50%
50%
Yes
No
Do you prefer to use a food recommendation system to plan what you cook for the next meal?
60 responses
30%
20%
50%
Yes
No
May be
In which form you prefer the recipe to be in?
60 responses
55%
16.7%
28.3%
Text
Audio
Video

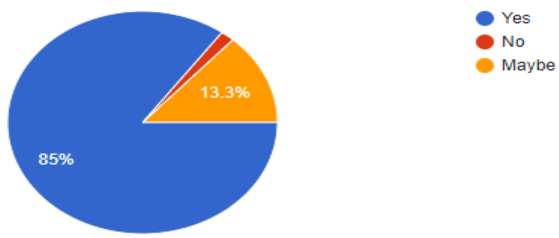
You are a

60 responses



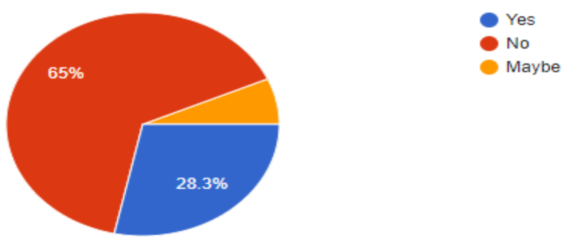
Do you encourage this application to maintain your personal diet

60 responses



Do you have personal dietitian or Caretaker?

60 responses



Are you aware of what food you should take for your diagnosis?

60 responses