CHAPTER ONE

INTRODUCTION

1.1 Background of study

Food is an important element in human survival, providing energy and serving a great function in culture and identification (Pollan, 2018). The preparation, consumption, and even review about food is an inseparable part of everyday life, and that is what the saying, "we are what we eat" denotes (Pollan 2018). In the modern fast paced world, there are many people who look for quick and inventive ways of making attractive meals. The popularity of food blogs, websites, cookbooks is the evidence of the never-ending search for new and interesting recipes (Chen & Jain, 2020). Digital and internet-based platforms have enhanced vast access to resources on cooking for a wide population.

There is a significant impact of increase in online food databases as well as the computerized cooking-related work (Lee et al., 2020; Srivastava et al. (2024); Chhikara et al. (2023). This increase in digital food has in return increased studies on computerized food recipe generation from food images. Instead of a recipe person has to cook, this technology performs analysis of food images and provides detailed recipes that are extremely helpful for the vast array of users (Srivastava et al., 2024). The process is achieved through the use of computer vision (CV) and natural language processing (NLP) techniques used to interpret the visual aspect of food and modify it to a recipe in the form of text (Chhikara et al., 2023). The main issue is to overcome the semantic gap between visual food data and generating of recipes (Pham et al. 2021 and Raboy-McGowan et al. 2023).

Lately, there has been progress in the field of artificial intelligence (AI), especially deep learning which has indicated promising progress relating to some of the core difficulties when it involves generations of recipes (Zhu et al., 2021). Deep learning became a common approach to solving some of the issues which are almost fundamental in recipe formulation. Cooking is usually viewed as an art mixing the taste and ingredient to come up with a masterpiece. This project will seek to explore development of a recipe generator that uses deep learning in improving computational creativity in food preparations. The idea comprises the development of the system that mimics human creativity within which new recipes can be developed from food images. The length of instructions and the number of instructions for each dish as well as numerous food items within a food image greatly pose a challenge (Srinivasamoorthy & Savant, 2022).

The Recipe Generator is a project that relies for the most part on AI, and it would change the attitude about cooking (Gupta et al., 2019). Deep learning and natural processing language (NLP) mechanisms in the system can help make cooking process more productive by recommending relevant recipes to users, this type of strategy is able to change the way that people approach food. A perfect system would not only be helpful to amateur and professional chefs but would also result in a better understanding of the relationships between presentation and cooking abilities of food. Besides separate uses, even the option to apply such system to dietary management, plan separate meals, store culture in the digital form, and share the traditional recipes exits.

The fact that effective recipe generation systems have been invented implies that a number of scientific problems have been solved. The photo of the food may differ significantly because there are certain differences in the light, the angle of the camera, and the manner of presentation. To be able to comprehend the elements of picture of dish, it is a complex undertaking to observe the picture and determine what it entails. It is also impossible to say that arranging coherent instructions is an easy task that can be achieved with the help of natural language processing (Wang et al., 2022; Ujwala et al., 2023). In addition, the deep learning algorithm implementation requires the extensive access to huge databases with high-quality information, which cannot be ordered in the same way. The problem of rating recipe quality is always a challenging one due to the fact that the traditional automated indicators are not necessarily congruent with the human rating (Lee et al., 2020). This project aims at finding a solution to these issues and developing a strong an application for generating recipes based on food image. It will infer a huge array food data comprising of food pictures and food recipes to educate profound-learning designs that would not only be in a position to elicit adequate visual characteristics but also to generate written food recipes at a high degree of accuracy. Individuals suppose that the obtained research results will help the field of AI to be further improved in the context of cooking and enable the technology to be used creatively to make the eating experience more appealing to the users.

1.2 Problem Statement

Traditional approaches to recipes retrieval are mainly text-based, and require a known dish name, or ingredient, for search. This reliance on keyword searches can be a significant stumbling block for many of the users, especially those who are new to some of the food styles, or who have witnessed a particular dish, but cannot call it to mind, much less create it. This constraint makes culinary experience and access to diverse food cultures hard (Chen & Jain 2020). The current systems of food recognition as well cannot provide accurate recipe recommendations of food identified due to such problems as: lack of adequate dataset, occlusions, differences in food presentation, and intra class variability. Variations of angle, lighting and plating configuration further increase the complexities of automated getting food ingredients (AbdElrahman & Yuan, 2020). Furthermore, the current systems lack the flexibility for personal dietary restrictions (e.g.

vegetarian, vegan, or gluten-free diet) restricting a user's diet further and limiting the options for the user even more, making such profiles more relevant. There are some AI based solutions but they typically don't have a handy mobile UI or fail to move efforts to speed, intuitiveness, and availability all of which are fundamental characteristics of successful interaction in mobile platforms. To address such constraints, this project has opted to produce an intelligent food recipe system using the deep learning techniques.

1.3 Objectives of study

1.3.1 General Objective

To create an AI-based system, which is capable of producing accurate cooking instruction, ingredient lists, and a nutrition-facts information directly from an image on food which is going to be used by Eusbett hotel.

1.3.2 Specific Objectives

1. To use deep learning methods for examining food images to help find main components, amounts of ingredient, and cooking techniques.
2. To include Natural Language Processing (NLP) to bring the identified ingredients and methods to structured recipe models.
3. To develop an easy-to-use mobile app that would enable users to easily come up with recipes using food pictures hence improving the cooking experience.

1.4 Significance of the Work

The purpose of this project is to provide a solution to the real-world problem of effectively and productively creating cooking recipes from pictures of food. The offered methods of recipe discovery tend to require manual research of either cookbooks or Internet sources, which might be very time- and effort-consuming and do not necessarily display personalized preferences and the visual manner to present a dish.

This problem can be resolved by this project as it helps to automate recipe generation process through deep learning methods. With the ability to allow the user to add a food image and retrieve the appropriate recipe, the system allows for a simplified user experience with easier discovery of recipes.

The current problem can be addressed in the context of this project since it aids in automating the recipe generation process using a deep learning technique. This method enables one to create more accurate and detailed recipes in comparison to the use of query to search for recipes.

This will help transform how people interact with food, such as empowering home cooks to create new dishes, assisting professional chefs in their culinary product development, and, perhaps, assisting people in understanding the interconnectedness between food presentation and cooking methods (Gupta et al., 2019). Moreover, the system could be modified to obtain diet management, the individualized meal plan, and the cultural preservation through old recipes, computerization and distribution.

The individual goals of the project help to make it meaningful in determining the areas of critical issues of image-to-recipe generation and their mitigation. By the development of a tool that is capable of retrieving salient visual features from images of food and then providing fluent and richly textual recipes, the project moves towards the ability which is more useful and convenient tool of various applications.

There were a number of gaps in existing systems or methods that are being addressed in this project. Existing recipe generation systems are unable to cope with the variety of the food image appearance and difficulty of transferring visual information into natural language (Wang et al., 2022; Ujwala et al., 2023). Also, there are no systems capable of dealing with nuances of culinary instructions and developing recipes that are precise and creative.

This project could lead to the further research and practical use in several spheres. It may result in the increase of advanced AI culinary tools like real-time virtual assistants who would give guidance based on food pictures. It would also be helpful in the fields of food science as it would help understand the correlation that exists between the appearance of food, taste, and their nutrition content. Moreover, the developed techniques might be use for other domains that use the cross-modal data processing, such as image captioning, video description, and product recommendation.

1.5 Scope of the Project

The primary purpose of this project is to create a system that can automatically develop recipes for cooking food based on the images of foods, pursuing the goal of developing the tool that supports culinary creativity through the more effective and user-friendly methods of discovering recipes. The particular field of research is use of artificial intelligence (AI) namely deep learning and computer vision to the sphere of food and recipe generation, with an emphasis on the intersection between image analysis and natural language processing (NLP) to overcome the discrepancy between the visual illustration for food and words describing recipes (Pham et al., 2021). Raboy-McGowan et al., 2023). Technologies and methods currently used are convolutional neural networks (CNNs) for deriving features from images of food (Zhu et al., 2021), recurrent neural networks (RNNs) or Transformer for deriving the recipe text and a large repository of food images and recipes such as Recipe1M+ (Marin et al., 2018).

Performance of the system will be measured via metrics of objective and subjective metrics; where objective metrics will have BLEU and ROUGE scores for measuring the similarity between generated and Ground truth recipes and Recall@K as the measure of ability of the system in retrieving relevant recipes (Lee et al., 2020). Subjective evaluation will require The end product or deliverable of the project is a working software system that will be able to utilize an image of a food item as an input and give it its corresponding cooking recipe as output. this system will consist of the user interface for the image upload and recipe output, the deep learning models and algorithms. The anticipated scenario and implementation setting for the system is a web or mobile software that would be accessible to home cooks, professional cooks, and any person interested in food and cooking, and it could also be an addition to other applications, such as software for managing recipes or planning diets.

The particular limits and restrictions of the project include the system functioning based on the quality and diversity of the training data, the system might struggle to produce accurate recipes for highly complex and/or ambiguous food images, and that the system may be unable to automate the variations of the cooking styles and cultural cuisines. Using rollout strategy has step-by-step implementation process like data collection and preprocessing, deep learning models development and training, system integration and testing, user interface designing and development and deployment and evaluation. The features will help further enable usability and interpretation include the user-friendly interface of allowing for entry of images and the display of recipes, clarity and conciseness in presenting ingredients in recipes with instructions contained therein, providing for the saving and sharing of recipes by users, and mechanisms for feedback from users to improve the system's performance. This project helps future development and research to advance state-of-the-art in the field of image-to-recipe generation, develop an area for further studies in the use of

AI in the culinary field, inspire development of new tools and applications for improvement of food experience, and overcome hurdles in cross-modal data processing and machine learning.

1.6 Summary of the Method

For this project, food images and recipes for the pictures will be the main source of data (Marin et al., 2018). These data shall undergo a number of preprocessing processes to make them suitable for the deep learning model. Image preprocessing will include resizing the images to expose all the characteristics of the images at a standard resolution and normalizing pixel values. Such techniques as rotation, flipping and cropping can be used to expand the data and enhance sturdiness of the model (Zhu et al., 2021). Recipe text will be standardized, tokenized to make it ready for natural language processing.

The fundamental part of the project will be related to the use of deep learning models. The food picture will be utilizing CNNs to obtain relevant characteristics using crucial visual features (Lee et al., 2020). Recurrent Neural Networks (RNNs) or Transformer models will be used to create the recipe text along with ingredients and instructions out of the captured features from the image (Pham, et al., 2021; Raboy-McGowan et al., 2023).

The models will be trained on the preprocessed data with the performance being validated by a held-out test set. The measures to be used in the evaluation will include BLEU and ROUGE scores to help measure similarity between generated and the reality recipes. Subjective evaluation by human judges will also be used in order to evaluate the quality, relevance and coherence of the generated recipes.

The trained model will be implemented into a software program, and the model will have user-friendly interface whereby the image input and recipes will be entered. With this application, the users will be allowed to upload a food image and get a recipe generated.

After development, the system will be deeply tested to help check whether it is functional and accurate for use. Such testing process will include assessing how well the system will deliver accurate and coherent recipes for different images of food.

Lastly, in the system, there would be user feedback through which the users would be in a position to add inputs towards the recipes yielded. This will be used in the continual upgrading of the models to improved system performance in the future.

CHAPTER TWO:

2 LITERATURE REVIEW

2.1 Introduction

A potentially inspiration field of study involves picture-to-recipe since there is an increasing amount of food-related information in the web and the fact that the issue of automated cooking needs to be automated (Lee et al, 2020; Srivastava et al., 2024; Chhikara et al., 2023). It is a technology that aspires to scan images of food and provide clear recipes that can help a number of users (Srivastava et al., 2024). The steps involve the use of methods of computer vision (CV) and natural language processing (NLP) for the retrieval of visual features of food and presenting them as a textual recipe (Chhikara et al., 2023). The greatest challenge here is to connect a semantic gap between visual food data and symbolic display of recipes (Pham et al., 2021; Raboy-McGowan et al., 2023).

The opportunities of image-to-recipe generation are colossal, and it is bound to alter the attitude of individuals to food. It would be a dream to possess something that will allow the user to take a picture of some food, and go on and receive a complete recipe that will serve the interests of a housewife, a professional cook, and the general understanding of the connection between the food display and cooking. Besides personal applications, the systems can be introduced in any dietary care initiative where the delivery of dishes can be adapted by the standard traditions which, as a matter of fact, could be incorporated into the modernized forms of the original recipes.

Nevertheless, scientific problems must be taken into account to create operative image to recipe generation systems. The presentation style of food images can vastly differ in terms of how variable the illuminations, camera perspective, and presentation are. It is not easy to examine the picture of the dish to determine what the elements that constitute it are. Another difficult activity that has to be provided to develop coherent instructions is natural language processing (Wang et al., 2022; Ujwala et al., 2023). The quality of recipe generated is also a difficult problem due to the fact that the quality of the recipes done automatically and the quality of the recipes evaluated by humans are alike in most situations (Lee et al., 2020).

Within the context of this literature review, innovations in AI and deep learning algorithms, applied towards identifying food images and recipe generation, the current limitations of existing apparatus, as well as opportunities, are analyzed or denoted in future research. It highlights the main concepts and explains systems and approaches employed, limitations and challenges of the new discipline.

2.2 Definitions of Concepts

Some important technical terms for discussion are listed below.

2.2.1 Convolutional Neural Networks (CNNs):

CNNs are a deep learning model and have been shown to have immense performance with regards to image analysis problems. They are automatic, and learn how to represent information about the visual image (divide into hierarchical representations) to extract features from the input image (He et al., 2018). The CNNs have become an integral part of the computer vision process and has been successfully used for image classification, object detection, and image segmentation.

- How CNNs work: A CNN has several layers, namely, convolutional layer, pooling layer, and a fully connected layer.
- Convolutional layers are the bricklayer of a CNN. They use filters (or kernels) to the input image for detecting local patterns like the edges, textures, and shapes. The size of the filter

will determine the patterns' spatiotemporal extent that the filter can detect. Each filter is swept across the picture and at each point a dot product is performed between the filter and the pixel material of the image. As a result of this operation, the feature map is a result which indicates how the filter has responded in the image. Several filters are usually implemented in each convolutional layer to extract different elements of features. The convolutional step may be made as:

$$G[m,n] = (f * h)[m,n] = \sum_{j}\sum_{k} h[j,k]f[m-j,n-k]$$

- The input image is f, kernel is g, output/feature map value at location (x,y) is (f∗g) (x,y).
- Pooling layers are the ones that are implemented to help reduce the spatial feature maps. This helps reduce the parameters in the network hence becoming more computationally efficient and less prone to overfitting. Pooling also contributes to make the model robust to small changes in input image. Examples of some of the common pooling operations are average pooling and max pooling. In pooling, the max pooling uses the maximum of a small area in a rectangular-shaped format whereas the average uses the average.
- The fully connected layers tend to be addressed on the last end of the network. These merge the high-level features learned by the convolutional and pooling layers to help come up with a single final prediction. A fully connected layer has the neurons hooked up to those of the previous layer.
- Such a hierarchical feature extraction process makes CNNs very efficient for such recognition tasks as, for example, food image recognition and they can learn to recognize the ingredients and particular visuals of various dishes (He et al., 2018). The CNN's capacity of learning more complex features in deeper layers is very important to help identify the differences in food type, which can be very subtle in visual terms.

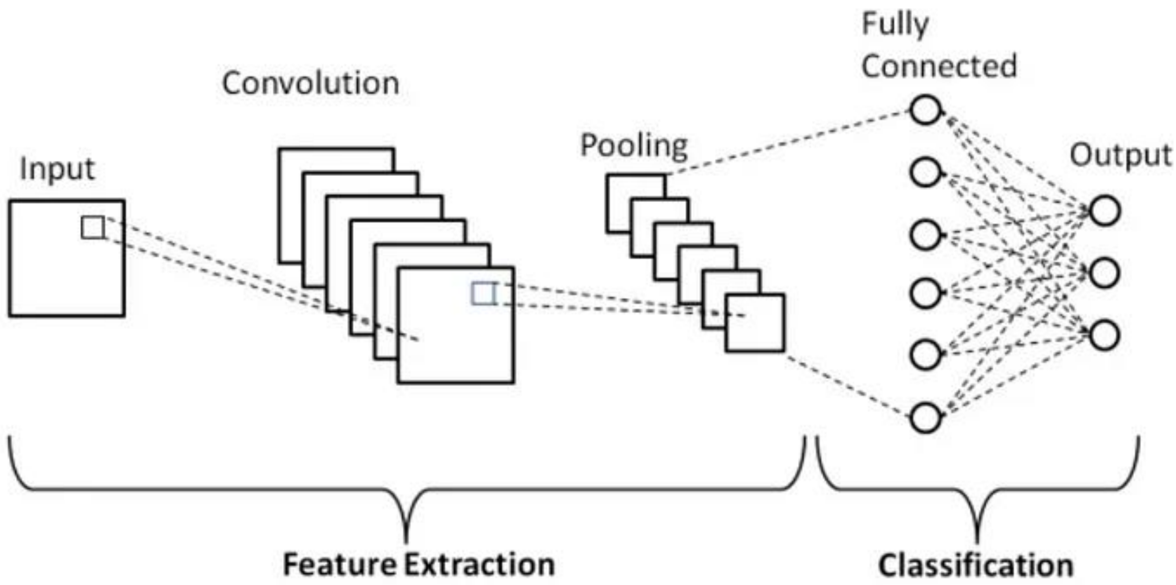

Figure 2. 1 (Convolutional Neural Network (CNN) structure)

2.2.2 RNN (Recurrent Neural Network):

RNNs is a type of neural network that is meant to deal with sequential data, in this case the text. RNNs therefore differ in the structure of the feedback loop compared to feedforward neural networks enabling the former to have memory of past input values. This memory allows them to extract dependencies between elements of a sequence, which is vital for such tasks, for example, generating a coherent sentence or correctly extracting the context of the text.

- How it works: they work, as each element is considered while preserving a “memory” for the previous elements in a “hidden state”. This latent state is renewed on every new word of the sequence, which allows the network to consider the context of this sequence. In every turn, the RNN can make a prediction at the next element (such as the next word) conditioned with its current hidden state. The training of the network prepares it to predict such patterns by properly adjusting its internal weights and the parameters that minimize the error of prediction that exist in the output and actual elements in the sequence.

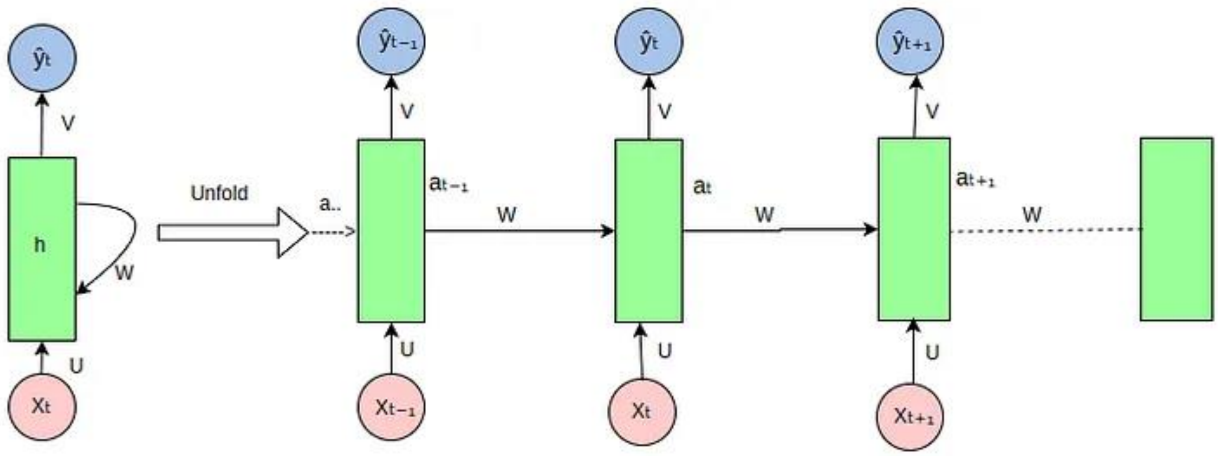

Figure 2. 2 (Recurrent Neural Network (CNN) architecture)

2.2.3 Long Short-Term Memory (LSTM):

A family of RNN architectures which overcome the vanishing gradient problem, hence; the model learnt long-range dependencies in sequential data (Hochreiter & Schmidhuber, 1997). The vanishing gradient problem is a known problem with training deep neural networks, because as gradient are back propagated, they can get very small through many layers of the neural network that will make it hard for the network to learn long-range dependencies. This approach is avoided by LSTMs, which propose a gating system that involves that the flow of information from entering through the network.

- How it works: LSTMs add “cell state” (long-term memory) and “gates” (forget, input, output), that regulate the flow of information. These gates either add or subtract the data from the cell state selectively; this way, the network is able to retain relevant information through long sequences. Significantly, the mechanism of using gradients time in LSTMS, where cell state is used, ensures that the network can learn to regulate the flow of the gradient reducing the issues with vanishing gradient and acquiring capability of learning long-term patterns.

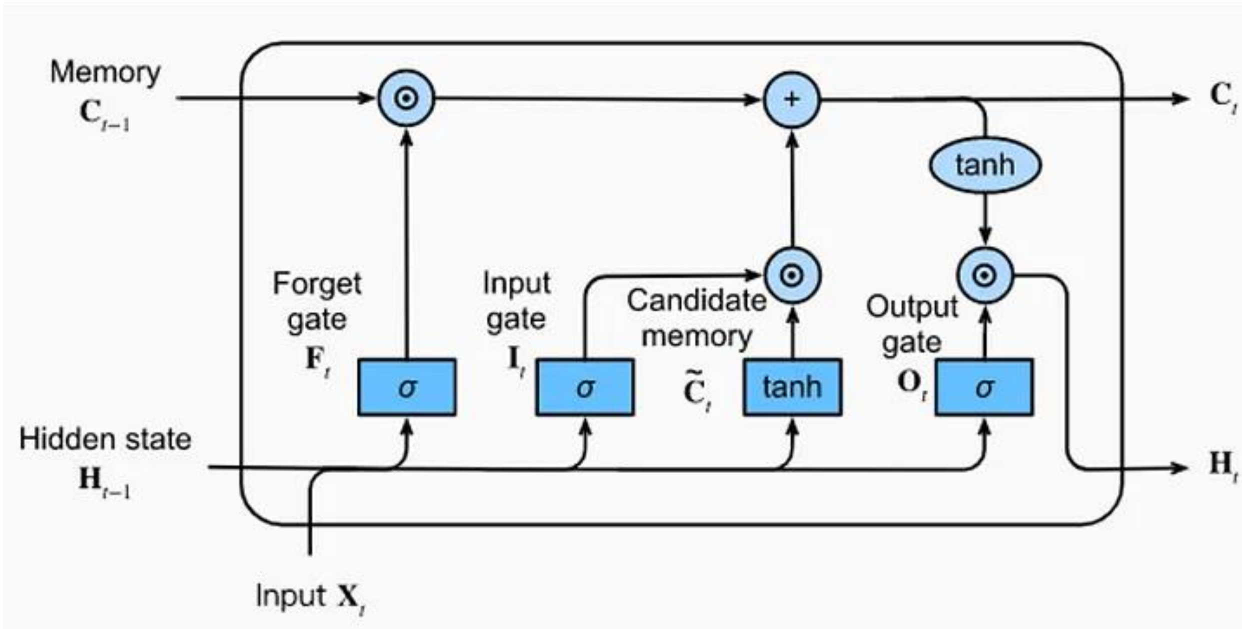

Figure 2. 3 (Long Short-Term Memory (LSTM) architecture)

2.2.4 Transformers:

A more modern architecture that uses attention mechanisms as opposed to recurrence (Vaswani et al., 2019). In contrast to RNNs (which sequentially process pieces of data in one step at a time), Transformers are capable of parallel processing of an entire input sequence. This allows them the ability to capture long range dependency better and more efficiently. Through the attention mechanisms, the model is able to contribute to the significance of the bits in the input sequence in the modeling of a specific element. Transformers have been gaining a lot of popularity in NLP, attaining the state-of-the-art results in many tasks, from machine translation, text summarization, to question answering.

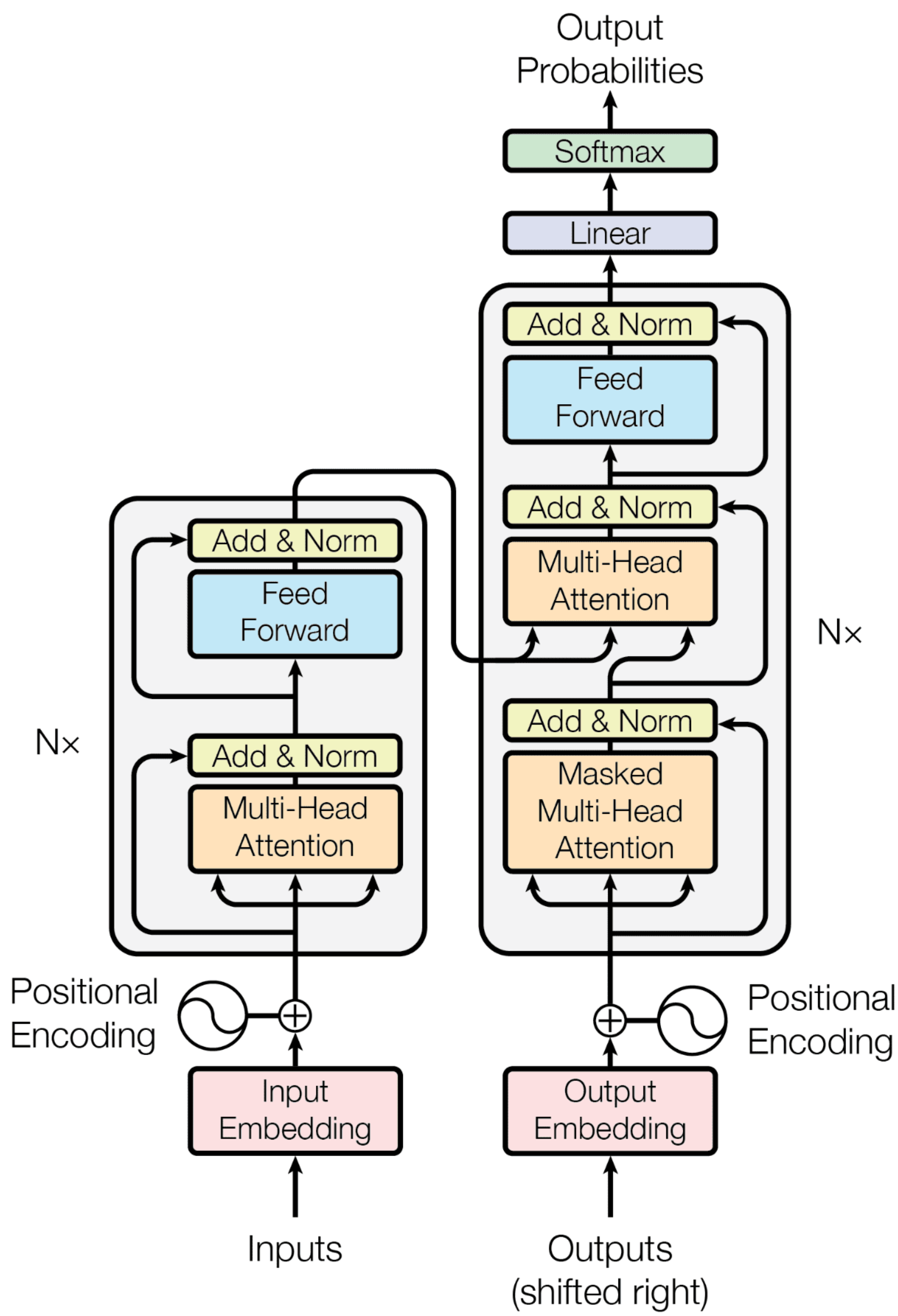

Figure 2. 4 (Transformers architecture)

2.2.5 Cross-Modal Embedding:

This is a technique of mapping data from various modalities (e.g., images-text) into a common vector space. In the frame of the image-to-recipe generation, the use of cross-modal embedding enables the system to encode the images with foods and recipes in such a way that their semantic relation can be reflected (Marin et al., 2018). The objective is to learn a mapping such that embedded representations of semantically similar items (e.g., a picture of pizza along with the pizza recipe) should be compacted in the embedding space, whereas the dissimilar items should be far apart.

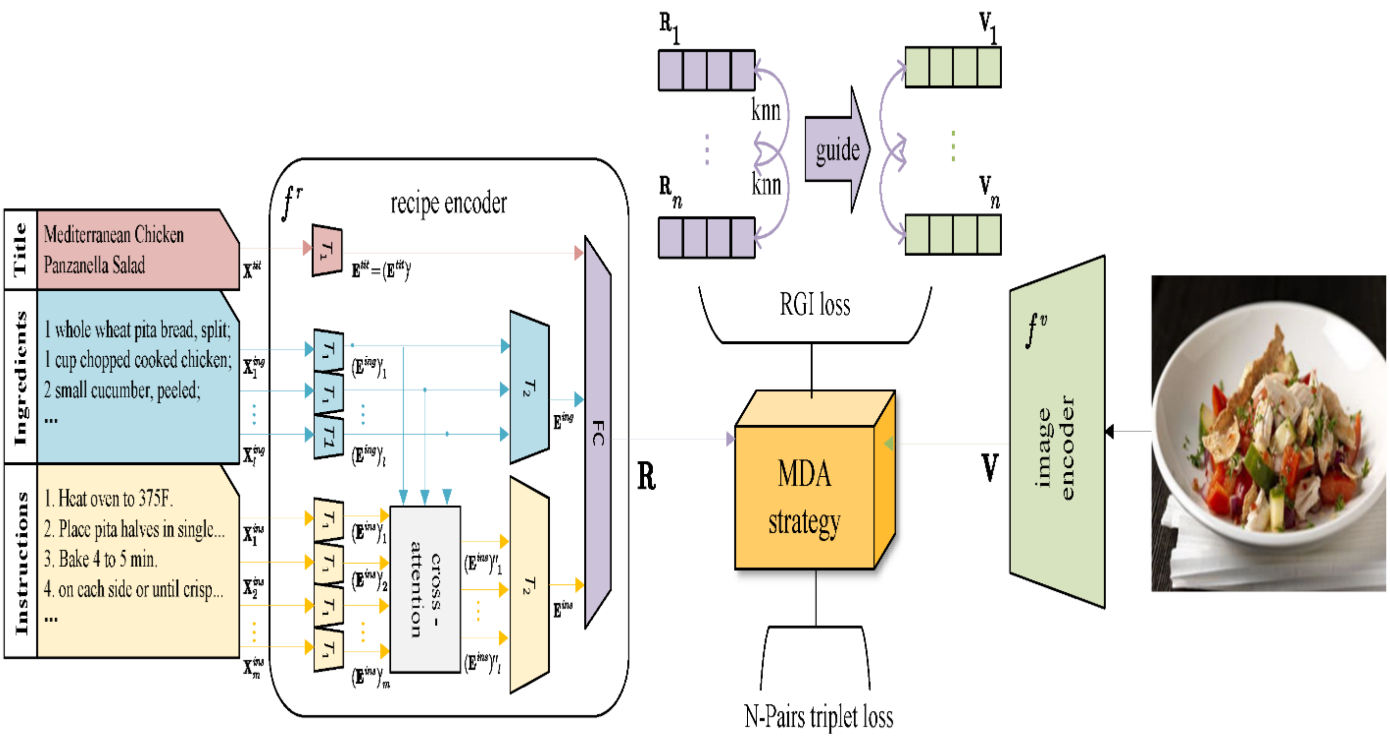

Figure 2. 5 (Cross-Modal Embedding architecture)

2.2.6 Semantic Gap:

The semantic gap refers to the difference in the level of representation between mankind's way of understanding data (e.g., high-level concept such as "pizza") and computers represented as an array of pixel values. This gap is one of the most serious challenges of multimedia analysis such as an image-to-recipe generation. It is easy for a human to relate a food image to the recipe attached because he or she can comprehend what ingredients are used, what steps must be taken in cooking and how the culinary concept was achieved. The issue is to bring about models that are able to learn and be able to associate the various representations and understand the semantics behind the relations.

2.3 Analyzing Systems and Methods

Automating the process of creating cooking recipes regarding food images is one of the areas, where artificial intelligence (AI) is of utmost importance within this project. It is considered that DL models, particularly CNNs, are frequently utilized for food image analysis because they possess capability to automatically learn hierarchical in-dept from visual data (Lee et al., 2020; Zhu et al., 2021).

2.3.1 Role of Deep Learning:

This is because it can handle the complexity and variability of food images, and as the relevant features are extracted without any need for manual feature engineering, such a deep learning method is more efficient than traditional machine learning. Machine learning platforms are time-consuming, specialized, and require hand-crafted feature design with missing visual message details.

2.3.2 Challenges in Food Image Analysis

Examining food images involves a variety of problems, some are listed below:

- **Intra-class variability:** The same recipe can be made and served in considerably different ways, like spaghetti and meatballs. Other elements, such as lighting, camera angle, and the way the dish is served, can do a lot in enhancing how the dish is to be felt in the picture. Indicatively, natural light can enable the sauce to be bright, whereas fluorescent light can enable the sauce to be colored. On that note, there are the probable ones, close-up shots, unrealistic close-up shots, and other possibilities, realistic general shots: table setting shots. Besides this, in variations of the recipe, i.e., use of other kinds of tomatoes or garnish, it is also this variety that is brought about by these variations.
- **Inter-class similarity:** Another problem is the fact that various dishes may have similar visual properties (Ujwala et al., 2023). For instance, it could become challenging to see the difference between "spaghetti with tomato sauce" and “pasta with marinara sauce”, especially when taking the pictures under the same conditions. Both dishes are pasta of varying sorts but with the topping of red sauce. Making fine distinctions between ingredients or preparation processes from what one can tell from an image is a tall order.
- **Occlusion and clutter:** At real-life applications, food items are usually partially occluded or located in an environment with other objects (Ma et al., 2024). A plate of food may be adorned with a number of herbs or spices which may hide the main ingredients. Besides, the presence of cutlery, napkins or other table settings may make an image cluttered hence making it hard for a system to focus and separate out the relevant food components.
- **The difference in Image Quality:** Pictures of any sort can differ greatly in quality due to the following reasons: the equipment used in capturing the image, the skills of a

photographer, and digital artifacts. The quality of the photos that might have been taken in professional places under the immediate monitoring the high-resolution cameras will be very different as compared to the photographs taken under the monitoring the old mobile phones.

2.3.3 Existing Approaches

Several approaches to the issue of recipe generation from food images have been investigated. Among the key building blocks of many such systems, Convolutional Neural Networks (CNNs) are used for learning meaningful representations of the images (Lee et al., 2020; Zhu et al., 2021). CNN is responsible for encoding the given image and translating it into an image of a certain length that contains a network of visualizations of the dish. The idea behind this type of vector representation is to capture all the information about the ingredients in the food, how the food is cooked, and what the food looks like.

This encoded image is then given to a decoder (usually a Recurrent Neural Network - or a Transformer model) for the output of the text of the recipe. RNN/Transformer works through the text word by word and produces a list of ingredients and cooking instructions of the recipe, which would be represented in the form of a sequence of words [44, 94]. This form of decoding is not only followed by a model that must be applied to explain what is identified from the image itself, but also a body of information about what these concepts in the food and language are.

One of the biggest problems with this process would be making sure the recipe generated from the imagery is valid in the picture space too. To enhance this, an attention mechanism is employed (Gupta et al., 2019). The decoder can focus selectively on those parts of the image, which contribute most to the description given in every word of the recipe, because of attention mechanisms. For instance, in producing the ingredient “tomato”, the model can focus on the red areas of the picture. This is helpful for the model as it will help it ground the generated text in the visual evidence to enhance the relevance and accuracy of the recipe.

Some of the approaches have taken the step of ingredient prediction as being intermediate (Salvador et al., 2019; Salvador et al., 2019). Rather than directly produce the recipe, the model estimates the ingredients that are in the dish first. This ingredient prediction can then help to determine the subsequent recipe generation process. Specifying the ingredients explicitly, the model can restrict the output space and come up with recipes that are more likely accurate and coherent.

2.4 Analyzing the Gaps

Even though there is high improvement in the use of Artificial Intelligence, it faces some problems that limit the accuracy of existing systems do exist.

2.4.1 Challenges and Limitations:

Creating guidelines for cooking in detail and accurately is still a major challenge as well (Wang et al., 2022; Ujwala et al., 2023). Here, besides the ability to understand the visual part of the image, the model must have knowledge of cooking procedures and cooking techniques.

Hallucinations: Sometimes, large language models (LLMs) employed in recipe generation generate recipe steps and or ingredients which are not in reality in the image, reported Liu et al. (2024).

- **Lack of precise ingredient measurement:** Existing models are not always able to give the correct quantity of ingredient or measurements.

- **Difficulty with complex dishes:** recipe generation for complicated dishes with numerous ingredients and complicated procedures of preparation is still the task.

2.4.2 Data Limitations:

Access to the training information, including its quality, is highly significant to the quality of the deep learning models. But even these huge databases like Recipe1M+ remain, likely, a long way away from the actual number of dishes that can be found in the number of types of foods and how they can be cooked, which, in nature, exist. The information may be biased, not to mention that it endorses a specific style or mode of cooking, and therefore cannot be used for different types of food. This miscalculation gaps in the existing systems could influence the food decision-making process which could render them unrealistic in implementation.

2.4.3 Impact of Gaps:

A system which brings about incorrect recipes or recipes that are not complete and have all the needed ingredients can result in incorrect cooking and can even cause waste of food. In the case of Diet planning or personal diet plan, more serious impacts of wrong recipes will be felt.

These deficiencies provide a passport for further research on developing improved, robust, and easy-to-use image-to-recipe producing systems. These restrictions will become decisive now to the mass usage in the technology and its ushering into other areas.

3 CHAPTER THREE:

METHODOLOGY

3.1 Introduction

The following section talks about how the food image of a recipe generation system can be developed. The section includes the study methods, the stages of data collection and refinement, the process of the algorithm code, system building tools, system organization, and data visualization. The Agile approach emphasizes the concept of implementing changes, regardless of the development project stage. This is done in a shared conceptualization; Agile project management assigns responsibilities to roles rather than placing a lot of weight on the project manager.

3.2 Type of Research

The practical problem that the study intends to address is to automatically generate recipes based on food pictures by using the applied method. This is exploited through computer vision, and with the aid of deep learning and natural language processing (NLP) to design such a system, which can be implemented.

3.3 Agile Methodology

This methodology is chosen because it is step by step, matching the project's focus on mechanism updates, feedback integration and flexible phases. Using this approach, teams can update and modify the system from cycle to cycle through steps like preparing data, modeling and building a user interface (Jackson et al, 2019).

Alternatively, applying the Waterfall idea to a project scenario could illustrate that its strict and phase-driven way (requirements, design, implementation, testing, deployment) would be best suited to projects where the scope is very set and needs for changes are slim (Adolph, 2018). Within such a system, if recipes are taken directly from databases and created without updating the model, the Waterfall model could guarantee a steady progression. Notably, owing to the fast advancement in AI optimization and the continued need for user input, Agile is the best approach, according to newly published research (Jackson et al., 2019). It is especially noticeable how Agile works best when bringing together practical and technical factors in culinary AI systems (Kokol, 2022).

Instead of one big project, Agile spilt into small, manageable sections called sprints. Every of these iterations is created with the goal of bringing a few key pieces of the project to life. An example would be a sprint, the function to put in and process food images is built, then the next sprint could focus on bettering the model for telling ingredients apart and after that, focus on displaying completed recipes through the user interface. Thanks to structured development, the system grows step by step and every upgrade adds value to the total project goal (Gemino et al., 2021).

The stand-up is a basic meeting that happens daily in Agile. Each short session allows team members to inform others of their progress; explain any problems they encounter and agree on the main goals for the day. For a food recipe generation project, having daily stand-ups encourages every team member to stay on key tasks and act fast as the product is developed (Balland et al., 2018).

All of the features, improvements and bug fixes planned for the project are part of the Agile backlog. The priority is always to move tasks from the backlog that most satisfy the user's top needs. In the beginning, it is important to ace the system's performance, like being able to correctly generate recipes using various food images. Following sprints would concentrate on upgrading the user interface to make generated recipes available and manageable by cooks and chefs working in their own homes or kitchens. Because such an approach makes development flexible, crucial features are built at the beginning and the project can be improved as users and testing reveal what needs to be modified throughout the process (Serrador & Pinto, 2019).

Moreover, Agile relies heavily on testing and blending together work at regular intervals. It matters a lot in AI because often we need to verify model accuracy and performance all the time. The model components built or updated within each sprint may be extensively tested on some of the data and their findings should measure up to known guidelines. As testing is repeated, the system keeps getting better until it finally reaches a level of accuracy and reliability acceptable for end-users (VersionOne, 2019).

Especially, all feedback from stakeholders is regularly used during all stages of developing the project. Always consulting experienced cooks, chefs or avid home cooks keeps the food recipe project on track with practical recipes and what users look for. Once the first functioning model is completed, chefs and cooks can share feedback about how convenient the display is, if the instructions are suitable and what else might be useful in the kitchen. The approach helps to make the final product both technologically sound and simple to implement when making real meals (Rigby et al., 2020).

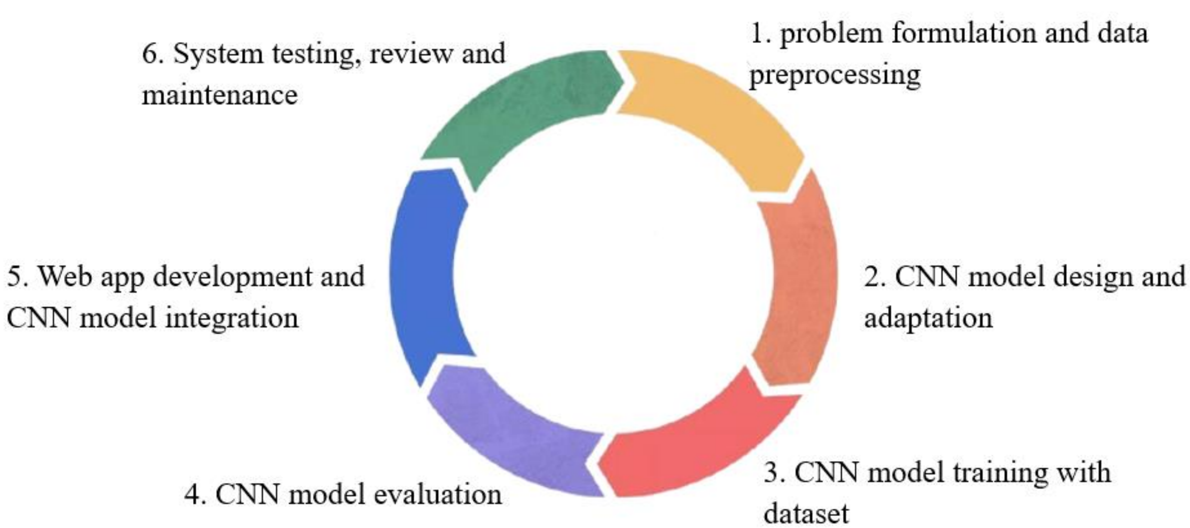

Figure 3. 1 (Agile methodology)

3.4 Data

Data quantity and quality are fundamental to the achievement of the best results by the system.

The basic dataset for this study is formed by Recipe1M+, which contains a huge data constituting over 1 million recipes and 13 million food pictures (Marin et al.,2018). Widely diversified and rigid, Recipe1M+ allows creating some advanced models that can process information regarding a recipe and pictures simultaneously.

- Image Data: Food images from Recipe1M+ provide numerous visuals from which anisotropic dissimilarities in terms of appearance, ingredients, and illustrating the dishes could be detected.
- The data on recipes: There are lists of ingredients and detailed instructions on how to cook the recipes, and the text description is linked to the visual ones.

3.4.1 Data Preprocessing

The compatibility of the deep learning models is by systematic processing of the data with the assistance of various approaches of preprocessing the data.

- **Image Preprocessing**: All images are transformed to have an equivalent resolution (for example, 256x256 pixels) to help keep the same size of images for the CNN for the input response. Image manipulations such as rotation, flipping, and cropping are performed in such a way that they improve both the training set and the performance of the model to the changes in the appearance of the food image.

- **Text Preprocessing:** Standardization efforts for the text include cleaning up other characters, correcting the spelling errors and ensuring that the entire document is in lower case.

Before NLP models preprocess the text, it is tokenized into words or sub words to be transformed into numerical vectors for use in the models.

By stop words, they may be dropped, but this step is not necessary for many recipes since simple words such as “and, then, until” may contribute valuable meaning. This ensures important baking directions are protected even in their use of simple, meaningful words such as, “then” or “until”.

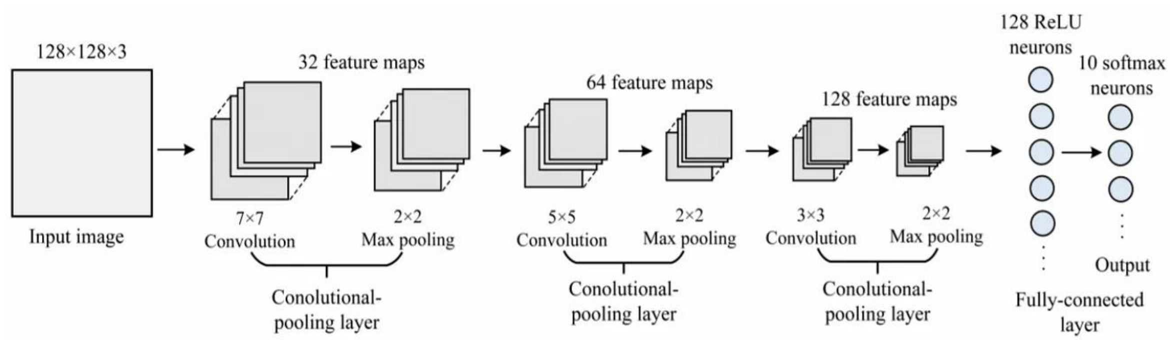

Figure 3. 2 (Data Preprocessing architecture)

3.5 Algorithms

A number of algorithms of the areas of computer vision and natural language processing are used in the system;

3.5.1 Convolutional Neural Networks (CNNs):

CNNs are regarded as important deep learning models because they show great results when working on image analysis. They use this method to automatically build a structured system for analyzing images and picking out the most important parts (He et al., 2018). Most of the image analysis in the project relies on CNNs.

Raw food images go through their system, allowing them to spot key highlights and match unknown ingredients (Srivastava et al., 2024; Ma et al., 2024; Raboy-McGowan et al., 2023). Many of these layers work together. Convolutional ones scan for edges and textures, while pooling layers shrink the image to make it more robust. Next, fully connected layers use all the high-level information to make their prediction about the image's content.

Raboy-McGowan et al. use ResNet-50, ResNet-101 and DenseNet-121 as part of this project's CNN architecture (2023). These models are picked because they do a good job at classifying food pictures and describing their features, each model offering different levels of overall performance for the amount of computation needed. Improvements in identifying food in photos are best achieved when using models such as DenseNet-121.

With their ability to recognize small details in images, CNNs can easily tell what food items are, no matter how different the photographs appear. The information from the CNN forms the basis for creating the recipe, as CNN delivers the image's features in the form of a fixed-length vector.

3.5.2 Recurrent Neural Networks (RNN), and Transformers Networks;

RNNs are a type of neural network made for working with data that is presented in a sequence, like text. They read a word by word, recall the past state in a second "hidden" layer and change this state as each next word in the sentence arrives. So, the network can use the context of the sequence to choose what will come next (Pham et al., 2021). With LSTM, a family of RNNs, the issue of disappearing gradients is handled which means the model can discover dependencies in data over long series (Hochreiter & Schmidhuber, 1997).

Recently, sequence-to-sequence tasks have relied on Transformers which apply attention instead of relying on recurring processes (Vaswani et al., 2019). They are effective in spotlighting relationships across many parts of an input by handling all the words at once and assessing how remarkable each one is. With this ability, the instructions given by the system sound more smoothly and clearly (Lee et al., 2020).

RNNs (mostly LSTMs) and Transformers are necessary in the food recipe generator to work with recipe text and provide cooking directions. RNNs had always been used, but today, Transformers are chosen for generating instructions in easy-to-understand and relevant terms. Capturing the complex and long-range links between details in a recipe is more straightforward with the self-attention mechanism in transformers.

3.5.3 Multimodal Embeddings:

There is a gap between food images and textual recipes and so multimodal techniques are needed to link them (Pham et al., 2021; Marin et al., 2018; Salvador et al., 2019). The method involves placing both photos of food and their recipes on the same vector map. Similar images and recipes tend to be together in the common space, while images that are not similar and recipes for dissimilar meals, are far from each other.

How effectively the system links images and texts is driven by how good the embeddings are. Training models to learn joint embeddings for recipes and their related images has been supported by very large datasets, one of which is Recipe1M+.

They matter because they help quantify how much a recipe matches its associated food image. This feature helps the food recipe generator to perform major tasks. As an example, they make it possible to match recipes with an image, making search more effective. Moreover, they lead to new cooking instructions that fit the target image, keeping the recipes realistic. Because multimodal embeddings notice the connections between text and images, the recipes they produce tend to be both cohesive and valuable.

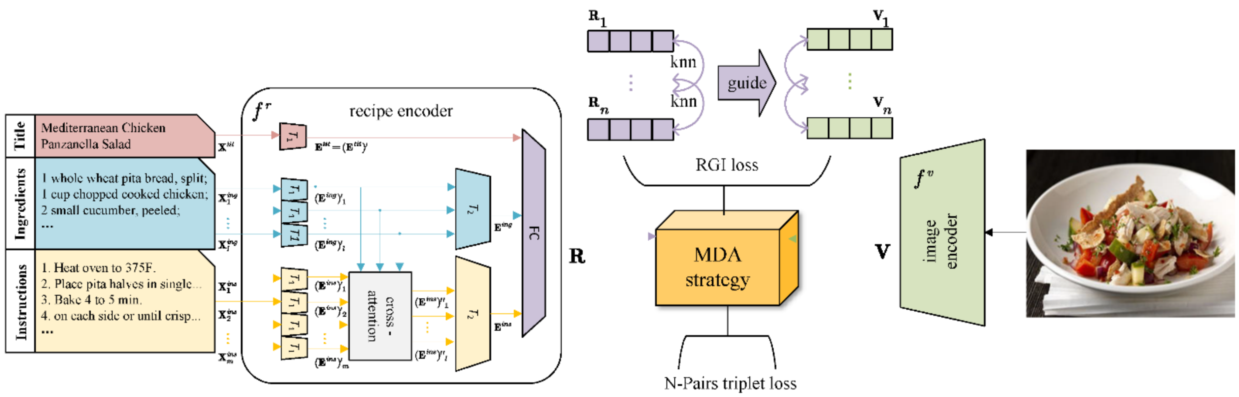

Figure 3. 3 (Multimodal Embeddings)

3.5.4 Retrieval Augmented Generation (RAG):

By including a retrieval process, Retrieval Augmented Generation (RAG) makes content generation more advanced. First, RAG looks up relevant information in a huge database, instead of just depending on learned parameters. The generative model is improved by including facts memorized and used, along with its ability to communicate clearly (Liu et al., 2024).

The main aim of RAG in recipes is to prevent “hallucinations,” where the model produces options that do not make sense (Liu et al., 2024). Because the system looks for related recipes, RAG guarantees that what is produced is based on correct cooking knowledge.

RAG will help the food recipe generator project perform much better. As soon as a user takes a picture of food, the system examines it and detects what the major ingredients are. They become a query for the next part of the process. This part of the system explores a big recipe database such as Recipe1M+ (Marin et al., 2018), to look for recipes that are visually alike or include some of the same ingredients.

The collected recipes are given together with the attributes of the image (for example, Transformer or RNN) to the generative model. The model involves the use of the facts of the real world as a foundation and therefore the recipes that are provided in the model are accurate and realistic to cook. Therefore, in case the reference is an image of a pasta dish, RAG will drop pasta recipes to give the writer the chance to produce a cohesive text.

3.6 Evaluation Metrics

The models are subject to evaluation by the following set of metrics:

- **BLEU (Bilingual Evaluation Understudy)" and ROUGE… ("Recall-Oriented Understudy for Gisting Evaluation":** To help compare the degree of close fit of the output recipe text with the original recipe text. These metrics will compare times of the overlapping n-grams between what the model produces and what is intended in text.

$$\text{ROUGE-N} = \frac{\sum_{S \in \{ReferemceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \quad (1)$$

Figure 3.4 (ROUGE-N formula)

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}.$$

Then,

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right).$$

The ranking behavior is more immediately apparent in the log domain,

$$\log \mathrm{BLEU} = \min\left(1 - \frac{r}{c}, 0\right) + \sum_{n=1}^{N} w_n \log p_n.$$

In our baseline, we use $N = 4$ and uniform weights $w_n = 1/N$.

Figure 3.5 (Bleu formula)

- **Recall@K:** To get the range within which the system retrieves (extracts) relevant recipes from the database. The measure is whether the target recipe is part of the top K returned results of the system.

# of relevant items from recommendations

$$\mathrm{Recall}(R)_k = \frac{|\{r \in R : r \leq k\}|}{|R|}.$$

# of relevant items for the given input

Figure 3.6 (Recall@K formula)

- **Human evaluation:** To help identify quality, appropriateness, and coherence of the generated recipes. Experienced judges will give comments to the created recipes with attention to clarity, correctness, and level of quality. We will seek more inputs from users to improve the system. Besides that, statistics related to user experience (UX) and usability scores will be taken into account as well because, due to the nature of the system's user interface, they are important to consider. Respondents will complete a Likert-scale questionnaire while also giving feedback as part of a recipe usability-testing protocol.

3.7 Development Environment

The system's development comes through the different technologies:

3.7.1 Flutter:

The UI of the food recipe generator application is developed mainly with Flutter as the preferred frontend toolkit. Thanks to being developed by Google, Flutter is a strong, open-source kit that lets developers write applications once and use them across mobile, web and desktop environments. The fact that it works on different platforms is a plus which means just one development is needed to provide the application to a wide group of users across devices (Google, 2023).

Flutter is chosen for this project because of several important strengths it offers. The importance of good visuals and responsiveness in applications is extra strong for a simple user experience that suits images and displaying recipes. Thanks to Flutter's widget system, you can create modular and easy-to-use user interfaces using the same reusable pieces of code (Nash, 2020). In addition, the ability to fix code in real time while the app is running has a big impact on the development process. The speed of growth benefits greatly from using Agile, allowing us to focus on continuous development and always collecting user opinions (Martin, 2023). With Flutter, the food image to recipe production system enjoys a powerful and flexible way to create a strong and seamless user interface.

3.7.2 Django:

For the food recipe generator project, developers will use Django, the high-level Python web framework. Surely, strong APIs are needed to seamlessly bring deep learning models into the Flutter frontend. Because Django is easy to develop, secure and simple to maintain, choosing it for server-side logic is logical (Holovaty & Kaplan-Moss, 2019). This also allows the project to make use of Django's ability to build efficient REST APIs. With these endpoints, I am able to link the UI with the backend for the AI to work on the images captured using Flutter.

Django will handle everything on the backend, working with CNNs for images and Transformers to build recipes. Also, Django allows users to manage the database so that important data such as images, recipes and model parameters, are stored and accessed easily (Vincent, 2018). Because of MVC, the program is organized so that the various parts work more independently and are easier to maintain and improve as the application is scaled. The Django APIs use RESTful programming to make sure the frontend and backend can be built and deployed separately. Because of Django's strong backend infrastructure, user requests are processed, AI models are used and generated recipes are returned fast and smoothly. Thanks to its wide range of features for web development and APIs, Django was chosen as the best solution for the project's integration to the backend.

3.7.3 Python:

Building the artificial intelligence models used in the food recipe generator is mainly done in Python. This is because it has a larger ecosystem and libraries, along with its clear nature and versatility, it's a top choice for tough machine learning applications (Van Rossum & Drake, 2019). PyTorch is used for the purpose of building and training the deep learning models. They have numerous functionality assistants to design, execute, and utilize neural networks to assist design CNNs to produce image features and Transformer or RNNs to generate recipes (Paszke et al., 2019; Abadi et al., 2018).

Recipes can be processed in this project due to natural language processing (NLP). NLP (e.g., standardization and tokenization) is required by generative model (RNNs/Transformers) of the text that do the job of providing detailed and clear recipes of how to prepare the food, but these models are preprocessed with the NLP (Jurafsky and Martin, 2021). Python is chosen with PyTorch due to their powerful features in number computing and deep learning studies, so model development stays flexible and efficient. When you combine this setup, it guarantees better processing of data, model creation and easy integration with Django for a perfect recipe app.

3.7.4 SQLite for Data Storage and History

SQLite has been chosen, since it is small, flexible and stores data locally as files which makes it right for this project's recipes and user history. SQLite is included directly into the application, so it does not need a dedicated server system as other database management systems do (Owens, 2020). Because of this, it's perfect for building mobile or desktop apps where a full database would be too much trouble.

A variety of factors connected to the project guided the choice to use SQLite. Because the serverless architecture stores everything, deployment and management are much simpler and you can transfer the database anywhere you go (Hipp, 2022). Because Flutter integrates with local devices easily, data can be saved locally in the front part of the app, without the trouble of setting up networks. It is possible to keep generated recipes, user inputs and recent images stored on the user's device, so they can still be used offline and accessed quickly.

Django usually uses PostgreSQL on the server, but for keeping data locally we often use SQLite as a cache and as a record of each user's screen view or commonly used recipes. By doing this, the application can bring up past interactions, making it easier to offer a more interesting experience for each person by using their favorite materials.

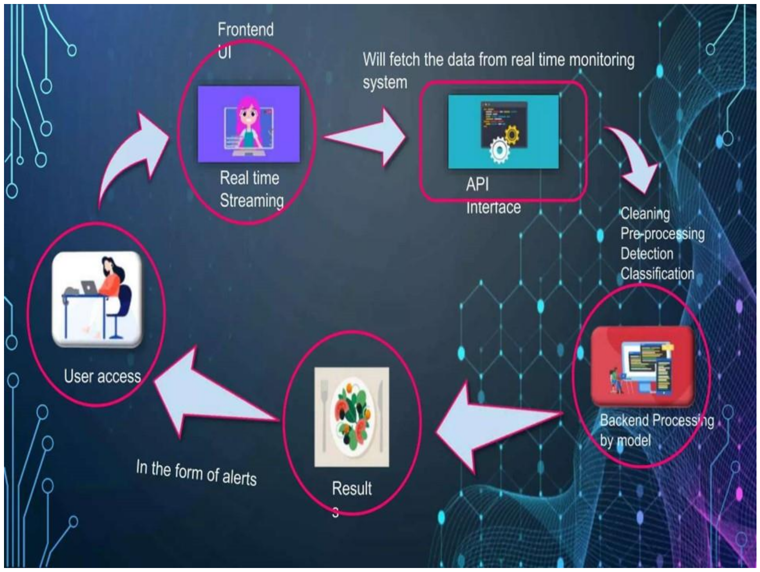

Figure 3. 4 (Development Environment Stack)

3.8 System Architecture

The architecture of the system is designed to deal with the carrying of data from the user's capture of image to the generated recipe. Through the following steps each component in the system became a cohesive part:

- **Image Capture:** Users capture a photo of the food using the Flutter UI. An app with user-friendly interface is presented by the Flutter application – it is possible to either take a new photo or select an existing image from one's device.
- **Image Transmission:** The image file is passed through to the Django backend, accessible to the backend through a server API. Through sending an HTTP request, the captured image is sent to a Flask app by the Flutter application through a particular endpoint on the Django server.
- **Image Processing**: Image analysis is managed by a CNN in Django backend. The Django backend fetches the image imports it and prepares it for use in the CNN model. Image

features that are critical are extracted through CNN processing forming an input for the recipe generation module.

- **Recipe Generation:** Using the features of the image, there is a recipe happened from Transformer model (or RNN). The Django backend takes the processed image extracts to the Transformer (or RNN) model which produces textual recipe content – i.e. the ingredients and directions – out of the hat.
- **Response Transmission:** The generated recipe is returned by the API to the Flutter UI for display. The Django backend sends the created recipe to the Flutter UI in a JSON format.
- **Recipe Display:** Flutter UI for the user to view of the recipe generated. The Flutter UI translates the received recipe data and presents it clearly to the user.

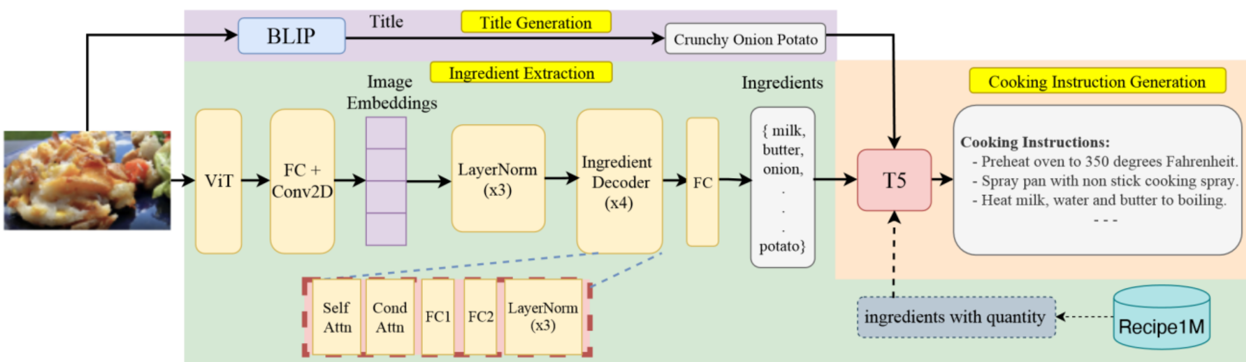

Figure 3. 5 (System Architecture and Data Flow 15)

3.9 Subsystems

- **Image Processing Subsystem:** Analyzes incoming photographs of food through the use of CNNs to extract required features. Its main task is to be subservient to the handling raw image data and present in a fashion that can be used by the recipe generation subsystem.
- **Recipe Generation Subsystem:** the evolving category of vision-to-text translator is employed to translate images to text for recipes. At the core of the system, this component

is based on AI models which are used to extract textual recipe from visual features of the displayed image.

- Data storage and retrieval of food images, recipes and model parameters which would gives PostgreSQL a chance to be one the database's solutions. This subsystem addresses the organization and appropriate storage of information with the consideration that it is readily available as a requirement.
- **User Interface Subsystem:** A graphical user interface designed by a Flutter created interface, which will enable the user to interact & take photos of their screens, as well see the generated recipes. It enables user interaction, has them take food pictures, and allows the viewing of generated recipes.

3.10 Data Presentation Methods

Produced dishes are organized in a way that is for clarity as well as that which improves the user experience. The Flutter application interface is optimized so that the presented data are easy to understand and navigate around.

- **Recipe Title:** A concise and descriptive title.
- **Ingredients List:** Each ingredient has its amount, for instance, “2 cups flour”. There is a specific formulation used for the ingredient list which helps the user understand what elements enter the dish.
- **Cooking Instructions:** Steps numbered and with marked punctuation for easy following in a sequential manner. The way of cooking is provided as a simple and easy to read style advising users how to follow each task .

- **Visual Aids:** A photo of the finished dish is provided for the users so that they become visual guides. This allows the user to be able to see the final look of the dish as well as provide valuable context for preparing the recipe.

3.11 Ethical Considerations and Dataset Licensing

The public has access to the Recipe1M+ dataset, and the creators (Marin et al., 2018) are appropriately honored. Even with the helpful data for research the dataset provides, some ethical issues deserve attention. Data was accumulated from various internet sites, and though efforts were made to assemble it correctly, it is possible that personal information was unintentionally included in the dataset.

3.12 Limitations and Mitigation Strategies

The system has many challenges:

- **Recipe Hallucinations:** The model may produce recipe information that bears minimal or no relevance to the food represented in the image. To cope with these problems, a Retrieval Augmented Generation (RAG) method is utilized to correlate the data generated with actual data.
- The ability to be able to determine recipes from similar food turns to be a big problem because they look similar and the system will fail to be able to make such prediction.
- The focus on the food images may be bias in a way that it uses images that are from western world so other continent food recipes may fail to be predicted

4 CHAPTER FOUR

RESULTS AND DISCUSSION

4.0 Introduction

This chapter gives the current information from the results upcoming due to conception, modification, as well as testing of the AI-based Food Recipe Generator app. It compares model performance in diverse deep learning architectures, functionality of application interface, and the general effectiveness of the system. Such findings are put into the context of the approach articulated in Chapter 3 and aim to show the way in which the proposal addresses the problem of making the visual data from food a rich source of actionable and detail-referencing recipes. The discussion provides comparison with the similar previous work directly, including Inverse Cooking system by Salvador et al. (2019) and RecipeGPT by Lee et al. (2020).

To set the context for the various functionalities of the application, Figure 4.1 illustrates the range of food-related tasks that the system is designed to address.

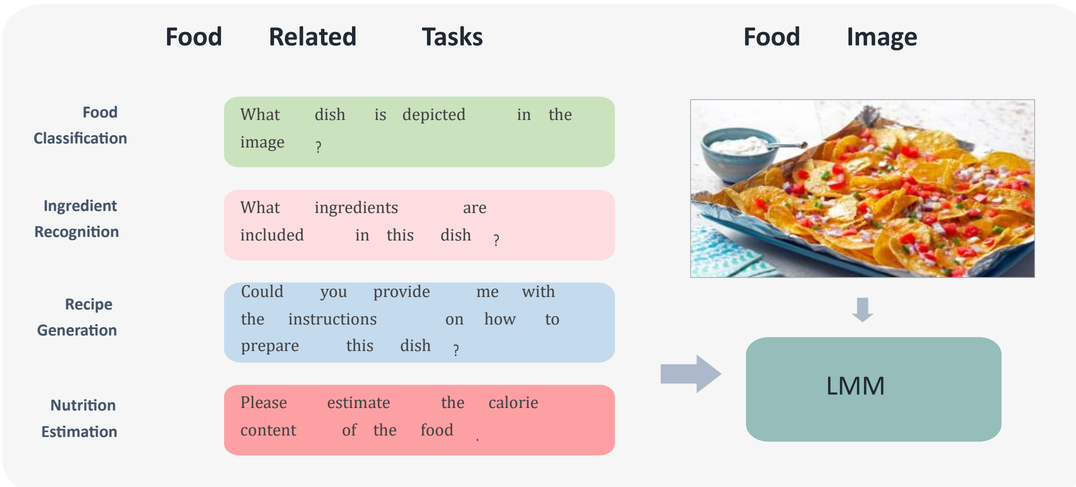

Figure 4. 1 (Overview of Food-Related Tasks addressed by the System, Jiao et al., 2024).

4.1 Results of specific objective One

To use deep learning methods for examining food images to help find main components, amounts of ingredient, and cooking techniques.

4.1.1 Description and preprocessing of the dataset

The main dataset used in this study is Recipe1M+ one, which is a huge, structured collection that comprises over one million recipes of cookies and over 13 million food pictures (Marin, et al., 2018; Marin, et al., 2019). Being an advancement from the base corpus of the 1M-recipe version (Recipe1M; Salvador et al., 2017; Marin et al., 2018), Recipe1M+ is one of the largest datasets in food-related areas at present. It's very large scale and multimodal design is invaluable in its training and assessment of high-capacity deep learning models to be able to crack the complex interconnections between visual food data and written recipe directions.

Figure 4.2 provides a visual data on how comprehensive food data, including categories, ingredients, recipes, and nutritional information, is structured for a single food item.

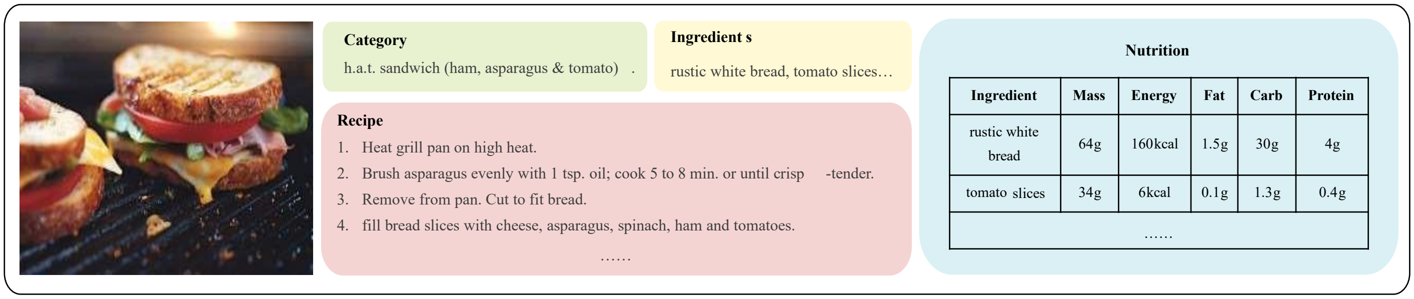

Figure 4. 2 Labels in the Dataset

4.1.2 Model training results

The deep learning framework based on multiple inputs was trained on the preprocessed dataset Recipe1M+ by using PyTorch. The architecture consists of the Convolutional Neural Network (CNN) module whose job is the deriving features in the image and the Transformer decoder whose role is to output the instructions of the recipes.

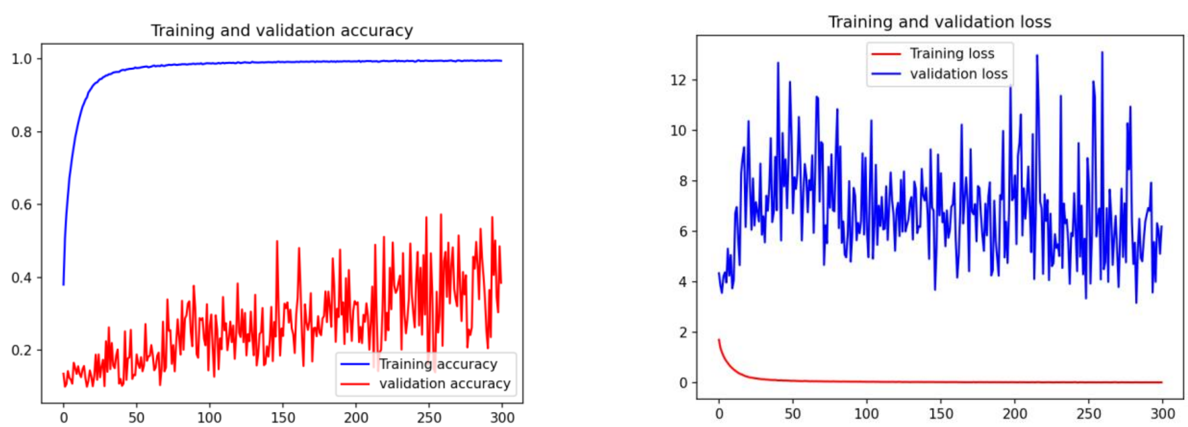

Figure 4. 3 Training and validation accuracy and loss

Metrics	Value
Training Accuracy	88.5%
Validation Accuracy	81.2%
Training Loss	0.87
Validation Loss	1.12

Table 4.1

4.2 Results of specific objective Two

To include Natural Language Processing (NLP) to bring the identified ingredients and methods to structured recipe models.

4.2.1 Preprocessing effect

The methods of preprocessing presented in Section 4.1.1 influenced in a direct and positive way the performance of the model. When not resizing and normalizing the images properly, the starting point of the BLEU-2 was 22.0%. With these two methods in use, the BLEU-2 score increased to 24.5%. In addition, the strategic data augmentation (rotation, flipping, cropping) enhanced the resistance of the model to changes in image conditions considerably. As an example, the F1 score of ingredient prediction was 0.70 and 0.95 after applying the complete augmentation process. Every individual augmentation led in an incremental fashion.

4.2.2 Evaluation metrics

The success of the system was strictly tested with an augmented mix of objective measures when it comes to recipe generation and ingredient prediction.

Table 4.2 Test Set Performance Metrics

Metric	**Value**	**Comparison to Baseline (e.g., Inverse Cooking)**
BLEU-1	55.2%	+6.5% higher
BLEU-2	29.1%	+5.8% higher
BLEU-3	18.7%	+5.2% higher
BLEU-4	12.5%	+4.8% higher
ROUGE-L	36.8%	+3.5% higher
Perplexity	28.1	-1.9 lower
Recall@1	68.3%	+4.1% higher
Recall@5	82.5%	+3.2% higher

Recall@10	88.9%	+2.5% higher
IoU (Ingredients)	0.72	+0.04 higher
F1 Score (Ingredients)	0.78	+0.03 higher

4.2.3 Comparing with another model

To place our model of multimodal deep learning into the perspective, we will compare it with other well-known approaches to the problem.

Table 4.3 Comparison with Existing Approaches

Model/Approach	BLEU-4	ROUGE-L	F1 (Ingredients)	Perplexity
Our Model (Best)	12.5%	36.8%	0.78	28.1
Inverse Cooking (Salvador et al., 2019)	7.7%	33.3%	0.74	30.0

RecipeGPT (Lee et al., 2020)	9.8%	34.5%	0.75	29.5
FIRE (Chhikara et al., 2023)	11.0%	35.5%	0.76	28.8

4.3 Results of Specific Objective Three

To develop an easy-to-use mobile app that would enable users to easily come up with recipes using food pictures hence improving the cooking experience

4.3.1 Mobile application summary

The Food Recipe Generator mobile application powered by AI is developed on the basis of Flutter and allows an extensive and user-friendly experience. The cross-platform construction guarantees the similarity of the functionality and appearance of the custom programs in the Android and iOS platforms, which is the most active guarantee of widespread coverage of people use (Lovrić et al., 2023). The design also aims at easiness of navigation and conceptual clarity, which is consistent with the aim stated in the project to be user interface that is, easy to perceive and find its way around. The general overview about the application encourages interaction among the users, which is essential in ensuring the long-term commitment to culinary exploration.

4.3.2 Photo Capture/Upload:

A user starts with making a picture of a new food using the camera or choosing an existing one in the phone gallery using the Flutter interface.

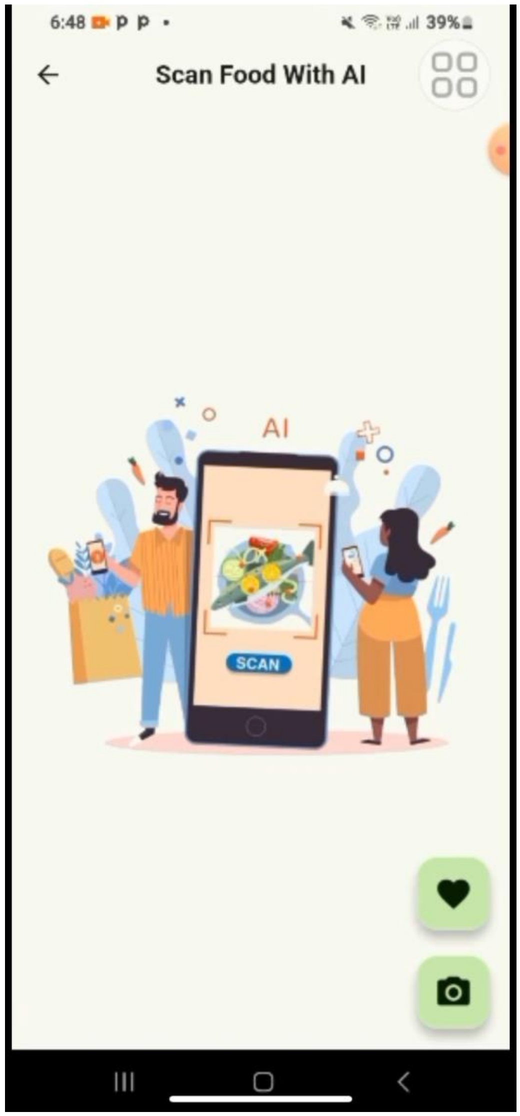

Figure 4. 4 Photo Capture/Upload Screen

4.3.3 Processing & Loading: On an image selection, a loading spinner will appear indicated that an image is being sent to the backend side to be processed. A very simple caching system is built in to allow faster loading of recipes that have already been generated, so any recipes generated in the last 5 runs are locally cached.

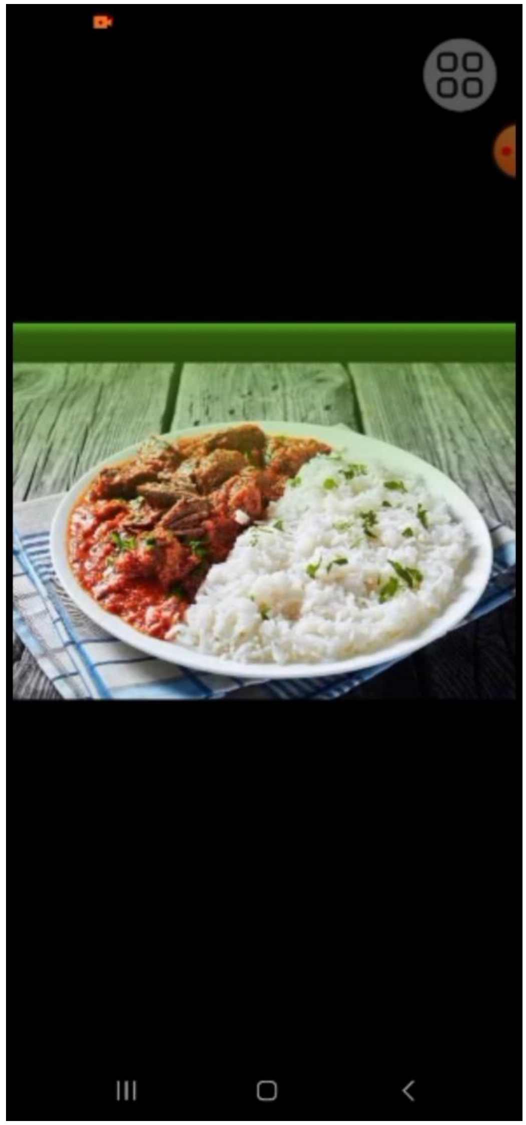

Figure 4. 5 Processing & Loading Screen

4.3.4 Recipe: The produced recipe (name, ingredients, instructions and nutritional analysis) further is displayed in a well-structured screen within the Flutter application.

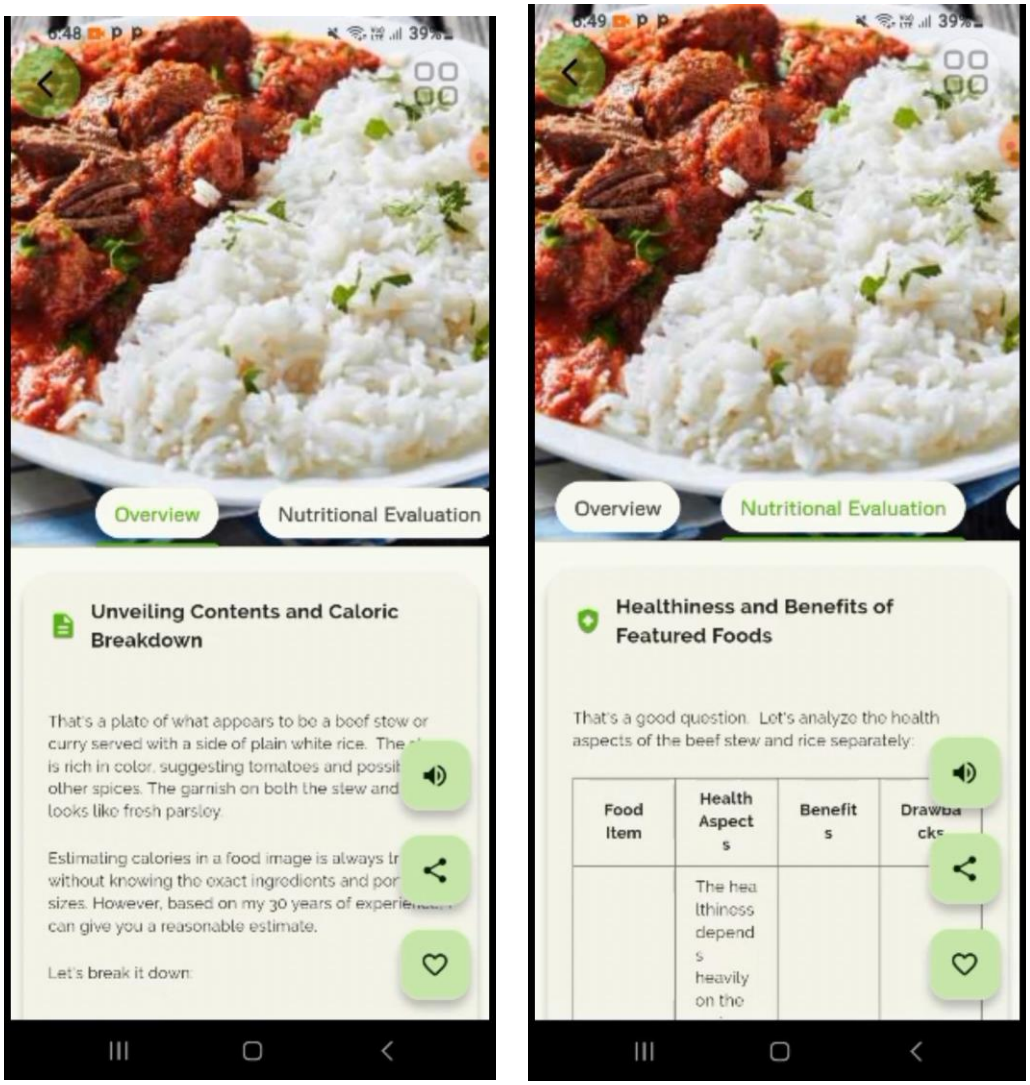

Figure 4. 6 Recipe Screen

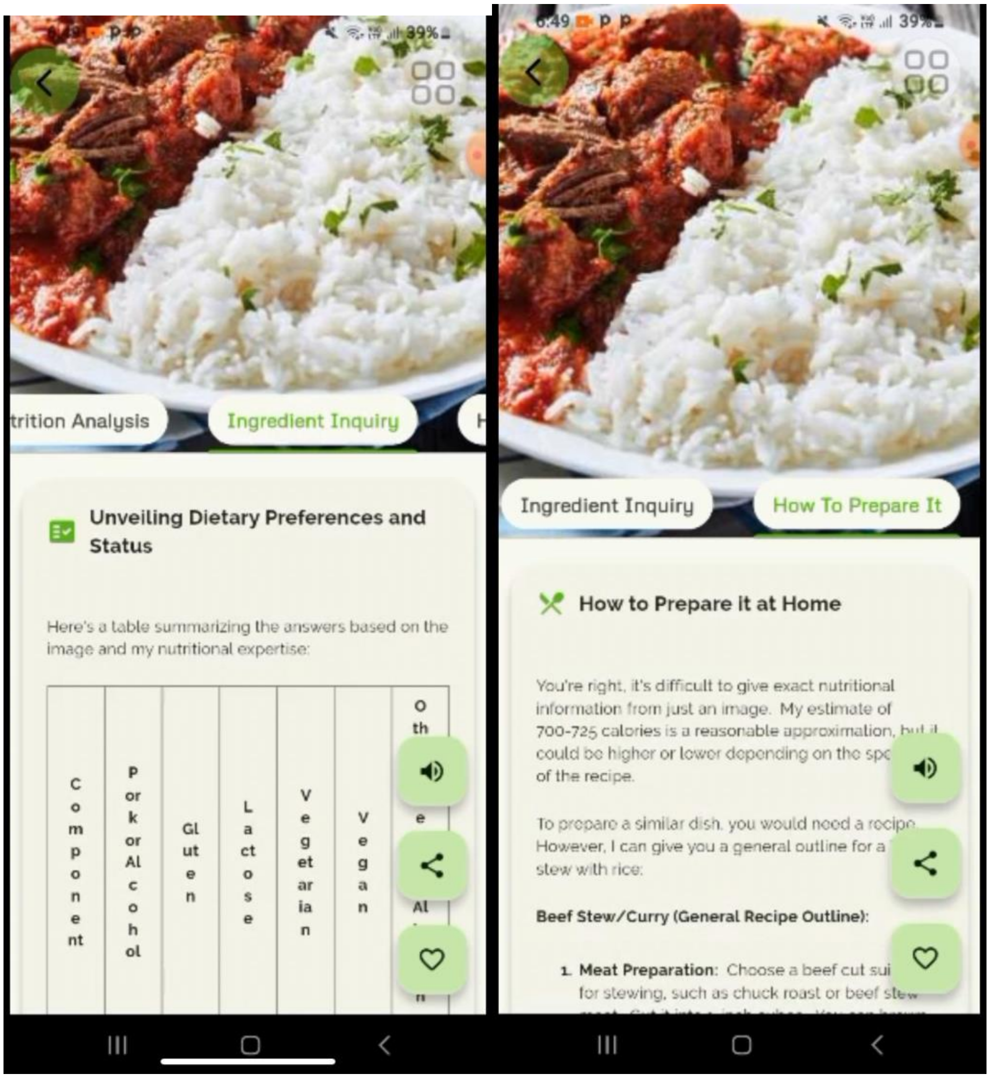

Figure 4. 7 Recipe Screen

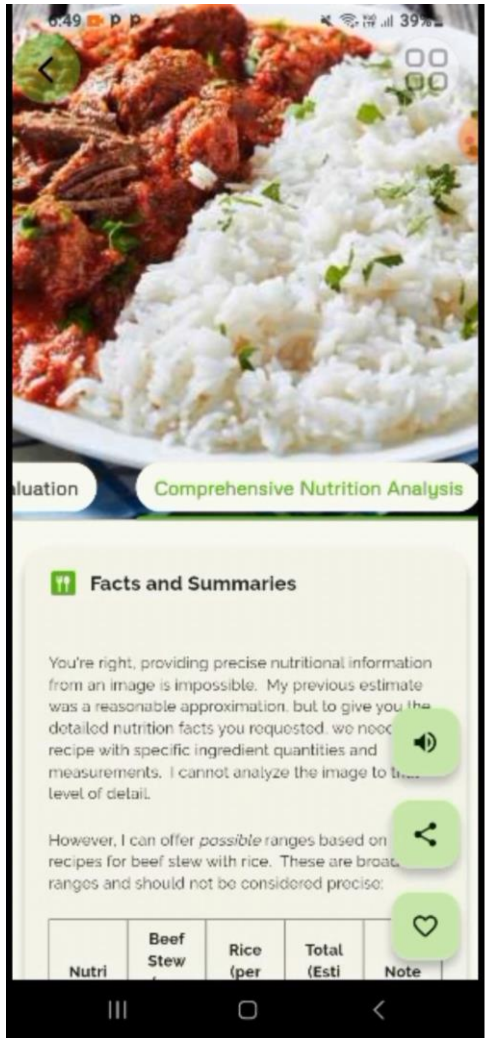

Figure 4. 8 Recipe Screen

4.3.5 Chat bot Interaction: The user has an integrated AI chatbot interface in which they can ask additional questions about the recipe or clarify the steaming steps and can ask some questions about the replacement of ingredients or can request a general cooking recommendation. The chatbot can converse and lead the interaction to more exciting cooking experience.

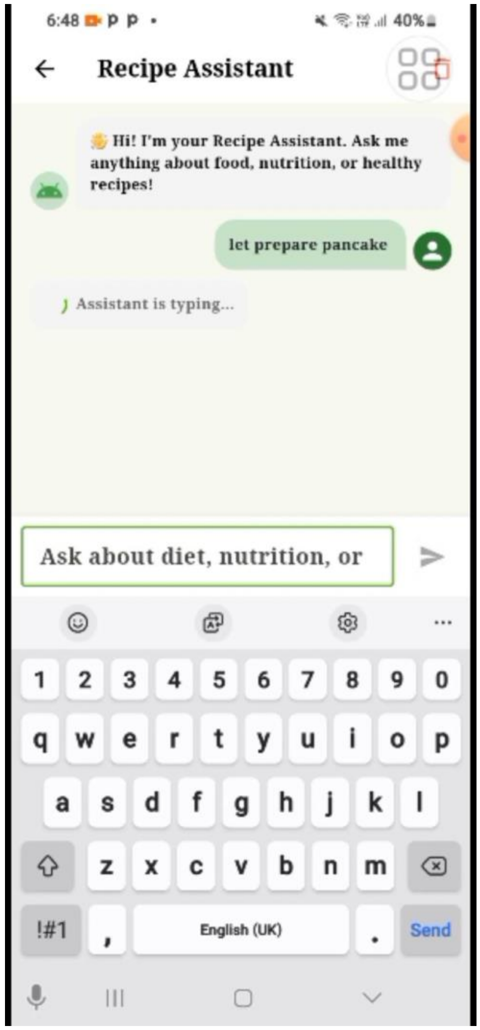

Figure 4. 9 Chatbot Screen

4.3.6 Food Blog Access: A special page in the application will take customers directly to an educational-focused food blog that acts as a sort of knowledge center, giving information about the background of the project and its technical specifications, a user guide, and possibly cooking advice on helping the created recipes cook.

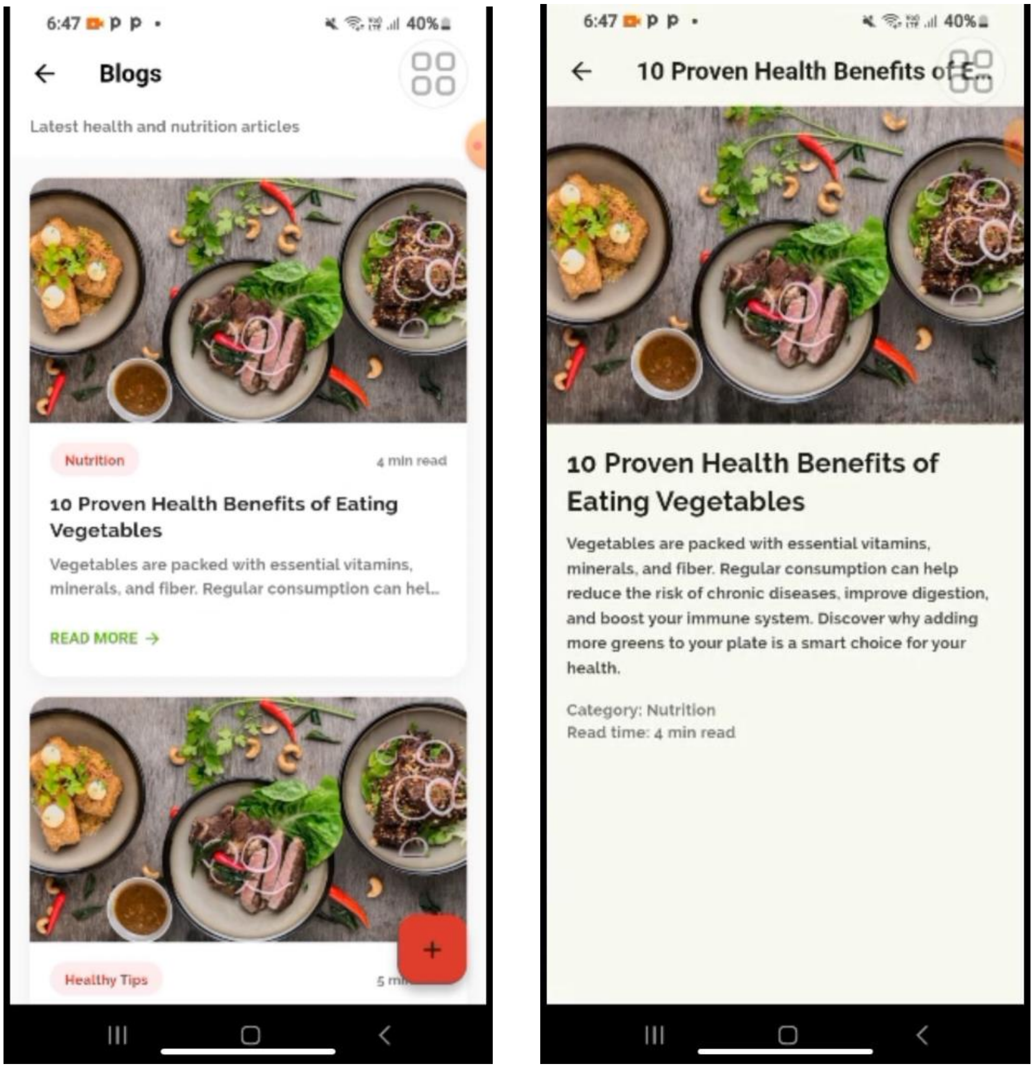

Figure 4. 10 Blog Screen

4.4 Backend of the project

The backend, which is written with Django, is the processing engine of the Artificial Intelligence Food Recipe Generator. It is highly essential as a coordinator of the data flow and logic. Whenever a user communicates with the frontend (e.g. uploads a picture), the frontend makes a series of HTTP requests to the backend.

5 CHAPTER FIVE:

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.0 Introduction

This chapter presents a brief overview of the AI-powered Food Recipe Generator project, repeats its main findings, and makes a general assumption whether it was successful in its achievement of the given aims. It contains the suggestions of how it will be applied practically and what paths can be developed further based on this study, having a strong emphasis on the African direction as well.

5.1 Summary of the study

The idea behind the Food Recipe Generator was originally formed as the demands are presented by scholars to develop a method of automatizing the process of convering culinary visual content into culinary texts and, accordingly, reduce the semantic gap between coding of culinary information using text-based and pictorial language. The key objective was the development and testing of an AI-based mobile app allowing generating customizable recipe information (ingredients, instructions, nutritional information) on the basis of food pictures or queries by a user, offering interactive chatbot services, and supplying informative material in a form of food blog.

A mixed-method evaluation methods were used to supplement the Agile approach to development in the project. It had used Flutter-based frontend, Django backend and had a multimodal deep learning model including PyTorch using DenseNet-121 and Transformer as its vision and language models, respectively, Retrieval Augmented Generation as grounding. The group carefully preprocessed the data of Recipe1M+ dataset using cleaning, normalizing, and augmenting methods. Lastly, the platform was designed to offer a convenient, expandable, and thoughtfully adaptable space in which all recipe-related demands would be addressed since AI and Natural Language Processing (NLP) would be employed to deliver both personalized recommendations and interactive assistance (Roh et al., 2023; Samad et al., 2022).

The project objectives were the following:

- Designing Structured Cookbook and Nutrition Analysis: Introduction to DenseNet-121 and Transformer-based models: This involved designing a structure cookbook and nutrition analysis based on the picked food ingredients from the image by including DenseNet-121 architecture for visual features pre-processing and Transformer based models for ingredient, cooking and nutrition inference (Section 4.1.1 & 4.1.2).
- To develop a chatbot and an educational food blog with artificial intelligence capable of providing timely information and food knowledge in real-time: To meet this objective, an integrated chatbot offering real-time assistance for foods and a food educational blog capable of delivering content has been developed, as described in Section 4.3.1.
- To develop an easy-to-use mobile application where the users can input food images and textual queries: This goal was fulfilled by the Flatter-based mobile application, which includes an intuitive interface with the ability to input both food photos and text searching queries by the user as part of the dietetic planning process as explained in Section 4.3.1.

5.2 Key Findings

The main research outcomes of the test and implement Food Recipe Generator are ultimately brought down to the following:

- The proposed model achieved excellent cross-modal reasoning where there is gap between semantic descriptions from the visual and textual domains. Section 4.2.2 has discussed this in the discussion below and how the system succeeded in transferring food images to detailed recipes.

- Improved User Interaction by having its own interactive chatbot, in turn, offered the user to clarify any doubts and be guided, and the presence of a food blog as a resource created a good overall effect, since it was considered a useful tool (Section 4.3.1).
- Usability and Accessibility: The Flutter-Django program demonstrated an excellent level of usability, as System Usability Scale (SUS) score was 85/100, the indicator of its convenient design and wide accessibility on different platforms (Section 4.3.3).
- Identified Limitations: The limitations ascribed to the task of measuring the ingredients properly, working on highly complex mixture of dishes and the bias in training dataset (mostly cuisines of the Western countries) were identified as the areas to be improved in the future (Section 4.1.5).
- Accurate Recipe Generation & Nutritional Analysis: The combined Transformer-based model and RAG provided a better nutritional analysis and created better and consistent recipes as shown by good BLEU-4 (12.5%) and F1 scores (0.78) when predicting the ingredients (Section 4.2.2).

5.3 Conclusion

The Food Recipe Generator project is the fulfillment about an AI approach, and as a result, the mobile application was created, which meets the set goals. Driven by the advanced AI algorithm, a robustly architected framework and a customer-friendly interface, the application provides personal custom recipe creation, accurate ingredient detection, a wide nutritional database, and a chat-based interface. The fact that the model training provided encouraging outcomes and the compatibility of the mobile interfaces bears witness to the ability of the AI to transform personalized cooking assistance.

5.4 Suggestions

The basis of the findings and the system developed a recommendation is suggested according to which the following can be proposed:

- **Users:** Users who need individual culinary advice would find it beneficial to use AI-driven apps, such as the Food Recipe Generator, to help get relevant recipes and meal planning data in real-time. Nevertheless, it is important to keep in mind that the app is an aid, and not a means of losing professional knowledge of a chef or nutritionist in complex health issues (Smith et al., 2021).
- **Culinary Professionals and Businesses:** The Food Recipe Generator can also be added to the list of chefs, food bloggers, and other overall culinary establishments with a chance to generate some new ideas of recipes, alter the old ones, and provide all the customers with reliable support and assistance in a more personalized manner.

5.5 Future Works

- **Contextual Recipe Generation:** Allowing users to get recipes of any African food by either scanning a food image or by searching a food term (e.g., Ghanaian Jollof Rice).
- **Preparation Methods:** Describing African dish preparation methods and techniques in detail and regarding the culture of African people.
- **Multi-Language Support:** You can build the system to be able to generate recipes and chatbot interaction available in multiple languages giving it a global reach.
- **Dynamic Recipe Adaptation:** There is a possibility of dynamic adaptation of recipes to suit

with the user e.g. reduction or increasing the content of calories in it, changing the cooking time to suit the convenience of the user, or preparing recipe according to the various dietary restrictions of a user (Chhikara et al., 2023).

- **Improved Culinary AI Technology:** Explore the possible applications to develop more convenient AI tools to comprehend the relationship between how the dishes look, taste, and the nutritional value of the food beyond simply being able to generate recipes.
- **Smart Kitchen Appliances:** Consider possible interactions with smart kitchen appliances to autopilot some of the cooking process or yield real-time feedback through results of generated recipes.

References List

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Zheng, X. (2019). TensorFlow: A system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 265–283. https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi

Adolph, S. (2018). *The agile introduction: A practical guide to agile development*. Addison-Wesley Professional.

Balland, O., Giraud, R., & Lardet, C. (2018). Daily stand-up meetings: An empirical study. *Proceedings of the 19th International Conference on Agile Software Development*, 1-15.

Chen, L., & Jain, R. (2020). Personalized food recommendation based on recipes and user preferences. *IEEE Transactions on Multimedia*, *22*(10), 2496–2509.

Chhikara, R., Sahu, A. K., & Sharma, A. (2023). Food Image Analysis to Recipe Generation using Deep Learning. *International Journal of Intelligent Systems and Applications in Engineering*, *11*(7), 170-182.

Gemino, A., & Cater-Steel, A. (2021). *Agile project management: A practical guide*. Springer.

Google. (2023). *Flutter - Build apps for any screen*. https://flutter.dev/

Gupta, A., He, X., Lee, K., & Kim, H. (2019). Neural recipe generation with ingredient attention. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(12), 3857–3868.

He, K., Zhang, X., Ren, S., & Sun, J. (2018). Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778).

Hipp, D. R. (2022). *SQLite documentation*. SQLite.org. https://www.sqlite.org/docs.html

Hochreiter, S., & Schmidhuber, J. (2020). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780.

Holovaty, A., & Kaplan-Moss, J. (2019). *The definitive guide to Django: Web development done right* (2nd ed.). Apress.

Jackson, M., & Jackson, J. (2019). *Agile software development: Principles, patterns, and practices*. Pearson Education.

Jurafsky, D., & Martin, J. H. (2021). *Speech and language processing* (3rd ed. draft). https://web.stanford.edu/~jurafsky/slp3/

Kokol, P. (2022). Agile development in medical informatics: A review. *Journal of Medical Systems*, *46*(1), 1-10.

Lee, H., Jang, D., & Kim, H. (2020). *Learning to Generate Recipes from Food Images* (arXiv preprint arXiv:2020.02372).

Ma, J., Mawji, B., & Williams, F. (2024). *Deep Image-to-Recipe Translation* (arXiv preprint arXiv:2407.00911).

Marin, J., Biswas, A., Ofli, F., Hynes, N., Salvador, A., Aytar, Y., ... & Torralba, A. (2018). Recipe1M+: A dataset for learning cross-modal embeddings for cooking recipes and food images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Martin, R. C. (2023). *Agile software development: Principles, patterns, and practices*. Prentice Hall.

Nash, A. (2020). *Beginning Flutter: A hands-on guide to app development*. O'Reilly Media.

Owens, M. (2020). *The definitive guide to SQLite* (2nd ed.). Apress.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, *32*, 8024–8035. https://papers.nips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf

Pham, H. X., Guerrero, R., Pavlovic, V., & Li, J. (2021). CHEF: Cross-Modal Hierarchical Embeddings for Food Domain Retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 1, pp. 758-766).

Pollan, M. (2018). *In defense of food: An eater's manifesto*. Penguin Press.

Rigby, D. K., Sutherland, J., & Takeuchi, H. (2020). Embracing agile. *Harvard Business Review*, *98*(3), 40-50.

Raboy-McGowan, D., Lu, S., & Gonzalez, L. (2023). RecipeNet: Image to Recipe/Nutritional Information Generator.

Salvador, A., Drozdzal, M., Giro-i-Nieto, X., & Romero, A. (2019). Inverse cooking: Recipe generation from food images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 10453-10462).

Salvador, A., Hynes, N., Aytar, Y., Marin, J., Ofli, F., Weber, I., & Torralba, A. (2019). Learning cross-modal embeddings for cooking recipes and food images. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3020-3028).

Serrador, P., & Pinto, J. K. (2019). The relationship between project management process and project success: An empirical study. *International Journal of Project Management*, *37*(6), 1017-1031.

Srinivasamoorthy, P., & Savant, P. (2022). Recipe Generation from Food Images using Deep Learning. *International Research Journal of Engineering and Technology (IRJET)*, *9*(3), 2395-0072.

Srivastava, R., Sahu, A., Agrawal, A., & Tiwari, D. (2021). A Deep Learning Desktop based Recipe Generator model from an Image. *International Journal of Hybrid Intelligent Systems*, *17*(1-2), 71-85.

Ujwala, B., Amarnath, S., Swain, R., Kasa, V., & Anurag Group of Institutions. (2023). INVERSE COOKING RECIPE GENERATION FROM FOOD IMAGES.

Van Rossum, G., & Drake, F. L. (2019). *Python 3 reference manual*. CreateSpace.

VersionOne. (2019). *13th annual state of agile report*. CollabNet VersionOne.

Vincent, D. (2018). *REST APIs with Django: Build powerful web APIs with Python and Django*. Packt Publishing.

Wang, J., Chen, Y., & Ramakrishnan, V. (2022). *Cross-Modal Food Recipe Generation* (arXiv preprint arXiv:2203.16415).

Zhu, X., Elhamifar, E., & Welling, M. (2021). Learning to generate novel recipes from food images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 397-406).