Self-Attention Architecture for Ingredients Generation from Food Images

Chaparala Jyothsna
UG Student, Dept. of CSE
SR Gudlavalleru Engineering College, Gudlavalleru
Andhra Pradesh – 521356, India
writetojyothsna@gmail.com

Dr. K. Srinivas
Associate Professor
SR Gudlavalleru Engineering College, Gudlavalleru
Andhra Pradesh – 521356, India
kalyanapusrinivascse@gmail.com

Bandi Bhargavi
UG Student, Dept. of CSE
SR Gudlavalleru Engineering College, Gudlavalleru
Andhra Pradesh – 521356, India
bhargavi2000bandi@gmail.com

Akuri Eswar Sravanth
UG Student, Dept. of CSE
SR Gudlavalleru Engineering College, Gudlavalleru
Andhra Pradesh – 521356, India
sra1theswar@gmail.com

Atmuri Trinadh Kumar
UG Student, Dept. of CSE
SR Gudlavalleru Engineering College, Gudlavalleru
Andhra Pradesh – 521356, India
trinadhatmuri@gmail.com

J.N.V.R. Swarup Kumar
MIEEE, Dept. of CSE
SR Gudlavalleru Engineering College, Gudlavalleru
Andhra Pradesh – 521356, India
swarupjnvr@gecgudlavalleru.ac.in

ABSTRACT

Food filming is becoming more popular among food connoisseurs. Each meal has a narrative that is detailed in a lengthy recipe, and sadly, just looking at a dish provides no insight into the preparation of food. There are various websites that aid in recognizing a meal by its components, however, no method has been successful at forecasting the ingredients in a dish. As a consequence, this research proposes a method for automatically generating the dish's recipe. This approach estimates the image's title and ingredients and then generates the image's featured meal's specific cooking instructions. This investigation looked at a range of Indian cuisines, including lunch and breakfast. To enhance user-friendly, this paper provides a web application that displays the recipe with a photograph of the dish.

Keywords: Image Recognition, Multi-label Classification, Self-Attention, ResNet-50, Recipe Generation, Food Images, Feed Forward Network

1. INTRODUCTION

Cooking is a pleasurable art form [17]. Nowadays, postings tagged with food and cooking have increased significantly on social media sites such as Instagram, Facebook, and Twitter. People found the effort of visualizing delectable meals and sharing them with the outside world to be enjoyable. Additionally, eating habits and culinary culture have evolved throughout time. Traditionally, most meals were made at home, but nowadays, many individuals consume food that has been prepared by someone other than their family or friends (e.g., online ordering, and restaurants). As a result, a system is required that lengthens the road to heat the meal, just as it did in the past.

Constructing a recipe from a food photograph seems to be a hard process. Because the components transform the cooking process. Generally, people can recognize the visible chemicals, and cannot always anticipate the substances that are completely dissolved in the meal (e.g., salt, pepper, flour, sugar).

The notion of image recognition is always evolving. However, this takes extra information about the meal, since coffee includes sugar and parotta is made with ghee. Thus, food identification requires existing computer image processing algorithms to go beyond the visible and include past knowledge to provide high-quality systematic descriptions of food preparation.

2. PROBLEM STATEMENT

Some several websites and applications provide assistance with the recipe and ingredients when the dish's title is searched. Numerous websites recommend a recipe based on the components that are given as input (e.g., dishes with leftover ingredients) and applications which classify the food using image recognition [8, 16].

Previously, suggested image-to-instruction algorithms were constrained by the datasets on which they were tested. They generate titles, ingredients, and recipes for the images included in the dataset. Their precision is limited to that particular space. When these systems are presented with a picture that does not exist in the static dataset, they often fail.

3. LITERATURE SURVEY

3.1. MULTI-LABEL CLASSIFICATION

The literature has made significant efforts to exploit deep neural networks in multi-label classification, including developing models and investigating loss functions that are well-suited for this job. One technique for extracting label dependencies is to use label powersets. Because powersets include all potential label combinations, they are unsuitable for solving large-scale issues. Another time-consuming alternative is to learn the labels' combined probability. To address this problem, probabilistic classifier chains and associated recurrent neural network-based counterparts offer to split the joint distribution into conditions, therefore establishing intrinsic ordering. Take note that the majority of these models demand that a prediction be made for every one of the possible labels. Additionally, joint embeddings of

the input and label have been established to retain correlations and forecast label sets. The process of multi-label image classification is shown in figure 1. When multi-label classification [11] goals are considered, binary logistic loss, target distribution cross-entropy, target distribution mean squared error, and ranking-based losses have been examined and contrasted.

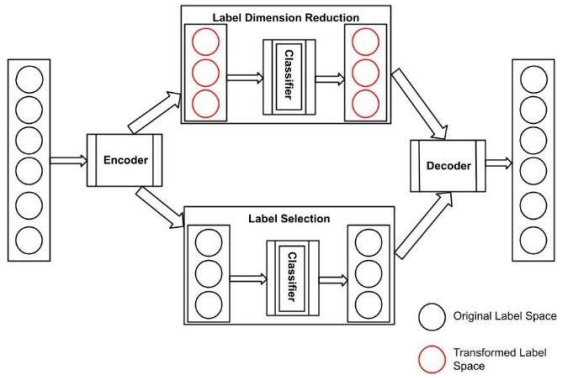

Figure 1: Multi-label Image Classification

3.2. CONDITIONAL TEXT GENERATION

Auto-regressive algorithms [10] have been widely studied in the literature, including both text- [2] and image-based conditionings [12, 13, 14]. The goal of neural machine translation research is to predict the translation of a given source text into a different language using various architectural designs (figure 2). Models such as convolutional and recurrent neural networks, as well as attention-based methods [20, 21] are all examples of these types of algorithms. Sequence-to-sequence models have recently been expanded to more completely accessible generation tasks, such as poetry and story creation. Using neural machine translation, auto- regressive models have shown promising results for picture subtitles, where the primary purpose is to offer a concise summary of the image's contents, opening the door to less constrained challenges like as creating descriptive lines of text or story-telling.

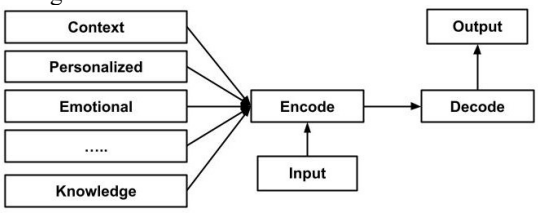

Figure 2: Conditional Text Generation

4. DATASET

The dataset for this study contains a variety of Indian foods, including breakfast and dinners. All of these images were gathered from a variety of cooking-related websites. The dataset comprises around 2500 photos of food, of which 1875 were used for training and 625 for testing [9].

As a consequence of scraping culinary websites, the recipes are much unstructured and often include redundant or highly detailed cooking items (e.g., olive oil, vegetable oil is separate ingredients). The lexicon of ingredients has a wide range of options because of the data being gathered from cooking websites (e.g., all-purpose flour, whole-wheat flour, bread flour). All of these substances were combined into a single category, significantly reducing the number of constituents. (For example, all-purpose flour, whole-wheat flour, and bread flour are all examples of flour). Finally, the dataset is whittled down to 127 components.

5. PROPOSED SYSTEM

The proposed system in this research resolves the issue of using a static dataset to generate recipes for food images. This image-to-instruction generating method forecasts the dish's title, ingredients, and cooking directions.

It predicts the ingredients from the picture and then generates instructions using both the image and the predicted ingredients. By examining the question 'does the order of the ingredients important,' the anticipated ingredient is viewed as both a set and a list. Additionally, this paper demonstrates using a limited collection of photos that food image-to-ingredient prediction is a difficult problem for humans and that our technique is capable of outperforming them.

The findings of this study may be stated thusly:

- An inverse cooking system is introduced that generates cooking directions from an image and its components while experimenting with different attention strategies for thinking about both modalities simultaneously.
- Furthermore, it provides a prototype ingredient prediction system that takes use of the co-dependence of components without imposing order.

6. METHODOLOGY

The system accepts a food picture as input and generates a recipe complete with title, ingredients, and cooking directions. Our technique begins by pretraining an image encoder and an ingredients decoder. The image encoder predicts a set of ingredients using visual characteristics taken from the input picture and ingredients. After that, train the ingredient encoder and decoder, which create title and instructions by putting the image's visual attributes and expected ingredients into a state-of-the-art sequence creation model [1, 5] (shown in figure 3).

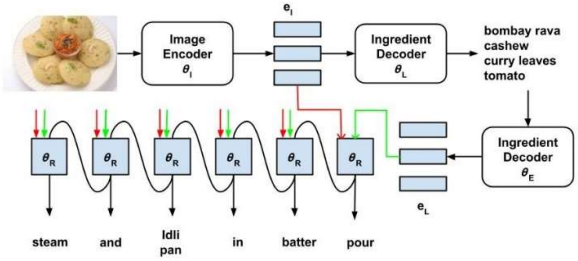

Figure 3: Process of image-to-recipe generation

After shrinking images to 256 pixels on their shortest side, the model uses a random clip of 224 x 224 pixels for training and a centre clip of 224 x 224 pixels for evaluation. Finally, for the instruction decoder, use a

transformer with 16 blocks and 8 multi-head attentions each with 64 dimensions. Later, use a transformer having four blocks and two multi-head attentions, with each 256 dimensions, to decode the ingredients. The ResNet-50 [24] model's last convolutional layer is utilized to create image embeddings. Both the photo and the component embeddings have size of 512 pixels. All dishes are limited in ingredients and directions to a sum of 150 words in this suggested format. Adam optimizer is used to train the models until the early-stopping requirements are met (with a probability of 50 and validation loss monitoring). Most of the models are implemented using PyTorch4. More specifics on the implementation are provided in the supplement.

6.1. ALGORITHM

Step 1: Input the image from the user.

Step 2: Extract the image features using ResNet. (Image encoder)

Step 3: Extract the ingredient using feed forward model. (Ingredient encoder and decoder)

Step 4: Generate the recipe using attention models [15] (using the image and the predicted ingredients)

Step 5: Display the title, ingredients and the instructions.

6.2. INGREDIENTS PREDICTION

As noted, before, the components prediction was accomplished in two methods. The first one stores the anticipated ingredients as a set, ensuring that no ingredient is duplicated in the final recipe formulation. The other is a list representation, which emphasizes the sequence of the elements.

6.2.1. LIST REPRESENTATION

1. Define a dictionary of N-dimensional ingredients as D

$$D = \{d_i\}_{i=0}^{N}$$

2. Create an ingredient list L by picking K components from D

$$L = [l_i]_{i=0}^{K}$$

3. Encrypt L as binary matrix of sizeK x N, with

$$L_{i,j} = \begin{cases} 1, & if\ d_j \in D \\ 0, & otherwise \end{cases}$$

4. Training process has M pictures, then the ingredient list pairing will be

$$\{(x^{(i)}, L^{(i)})\}_{i=0}^{M}$$

5. By optimizing the goal, predict $\hat{L}$from a picture x:

$$argmax_{\theta_I,\theta_L} \sum_{i=0}^{M} log\ p(\hat{L}^{(i)} = L^{(i)} | x^{(i)};\ \theta_I, \theta_L)$$

here,

θ_I, θ_Ldenotesparameters that may be learned

6. Because L is a list, so divide$p(\hat{L}^{(i)} = L^{(i)} | x^{(i)})$ into K conditionals

$$\sum_{k=0}^{K} \log p(\hat{L}_k^{(i)} = L_k^{(i)} | x^{(i)}, \quad L_{<k}^{(i)})^3$$

6.2.2. SET REPRESENTATION

1. Create an ingredient list L by picking K components from S

$$S = [s_i]_{i=0}^{K}$$

2. Encrypt S as binary vector of size N, with

$$S_i = \begin{cases} 1, & if\ s_j \in S \\ 0, & otherwise \end{cases}$$

3. Training process has M pictures, then the ingredient set pairing will be

$$\{(x^{(i)}, S^{(i)})\}_{i=0}^{M}$$

4. By optimizing the goal, predict $\hat{L}$from a picture x:

$$argmax_{\theta_I,\theta_L} \sum_{i=0}^{M} log\ p(\hat{S}^{(i)} = S^{(i)} | x^{(i)};\ \theta_I, \theta_L)$$

here,

θ_I, θ_Ldenotes parameters that may be learned

5. Considering that all components are independent, then quantize$p(\hat{S}^{(i)} = S^{(i)} | x^{(i)})$

$$\sum_{j=0}^{N} \log p(\hat{S}_j^{(i)} = S_j^{(i)} | x^{(i)})$$

Until the conclusion of the sequence (*eos*), the transformer anticipates the components in a list-like way. The downside of this technique is that it penalises for order, as previously indicated. It is recommended that the outputs be pooled across several time steps using a max pooling procedure in order to remove the sequence in which elements are predicted. [15].

Additionally, to guarantee that no component $\hat{L}^{(i)}$is picked twice, it needs the pre-activation of $p(\hat{L}_k^{(i)} | x^{(i)},\ L_k^{(i)})$ to be -∞ all of the previously chosen components at the prescribed intervals < k. This model may be trained by lowering the binary cross-entropy between the predicted constituents and the actual reality. By include the *eos* in the pooling procedure, the location of the token would be lost. So, account for an additional loss in order to determine the stopping point for ingredient prediction. The expected *eos* probability at all times and the ground truth probability at all times is the binary cross-entropy difference, which is defined as the *eos* loss. Additionally, it integrates a cardinality l_i penalty that has found to be effective experimentally. Then sampled straight out from transformer's result during inference. This kind is referred to

as a set transformer.

A second option is to model joint distributions of set elements by minimising cross-entropy loss between $p\left(\hat{s}^{(i)}\middle|x^{(i)}\right)$ and $p\left(\hat{s}^{(i)}\middle|x^{(i)}\right)$, respectively, and then use the target distribution $p\left(s^{(i)}\middle|x^{(i)}=\frac{s^{(i)}}{\sum_j s_j^{(i)}}\right)$, respectively, to model the joint distribution of set elements. Even so, it's not clear how to bring the target distribution back to a matched set of elements with configurable cardinality. The cross-entropy loss of the target distribution may be trained into a feed forward network in this scenario. In order to get a complete ingredient set, it is recommended to sample from a cumulative distribution of ordered output probabilities $p\left(\hat{s}^{(i)}\middle|x^{(i)}\right)$ and stop sampling when a threshold is crossed. Feed forward is a term used to describe this strategy (target distribution).

6.3. RECIPE GENERATION

Using an instruction transformer, the model seeks to generate a series of instructions [3, 4] $R = (r_1, r_2, ..., r_T)$ (where r_T is a word in the series of words. The first directive is the title [18, 19]. The image format e_I and the ingredient embedding e_L are concurrently processed by this transformer. It uses a ResNet-50 [25] encoder to get the image representation and a decoder design for ingredient prediction to get the embedding e_L, with a fixed-size vector assigned to each ingredient in the model.

Two attention levels are followed by a linear layer in each transformer block with in instruction decoding. Second layer of attention focuses on model conditioning in attempt to increase the self-attention [20, 24] outputs of the first layer. Transformer blocks are used to build the linear layer and the softmax nonlinearity in the model (figure 4), which provides a distribution of recipe words for each time step t. With the transformer model, it's easy to see why this is so frequently the case. This instruction generator, on the other hand, is constrained by two sources: picture features $e_I \in R^{P \times d_e}$ and ingredient embeddings (where K is the number of image and ingredient attributes that may be included into the embedding dimensionality). Since the suggested system requires [21] simultaneous reasoning about both modalities, the recipe generation process [6, 7] is being led by [6, 7] this ability. Three unique methods have been used to accomplish this goal, as shown below.

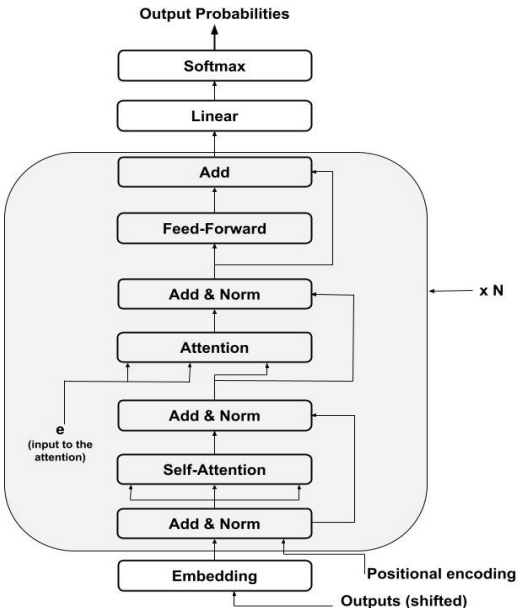

Figure 4: Transformer Model

6.3.1. CONCATENATION ATTENTION

This method combines the image e_I and the components that are predicted e_L in the very step of the dimension $e_{concat} \in R^{(K+P) \times d_e}$ (figure 6). Then, these embeddings are applied to the attention [1].

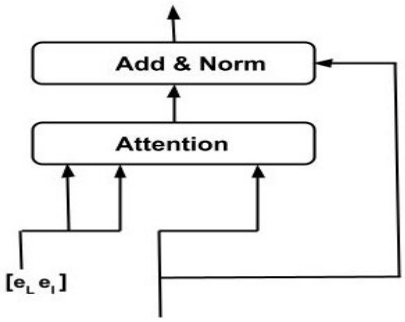

Figure 6: Concatenated Attention

6.3.2. INDEPENDENT ATTENTION

This method employs two levels of attention to cope with bimodal conditions. One layer is responsible for the picture embedding e_I, while the other is responsible for the ingredient embeddings e_L (figure 7). Through a summing procedure, the results from both attention [22, 23] layers are merged [1].

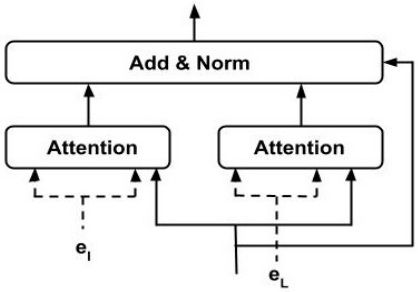

Figure 7: Independent Attention

6.3.3. SEQUENTIAL ATTENTION

This method alternates between the two conditioned modes progressively. As part of this study (figure 8), we look at the effects of attention being divided between two different groups: (1) those that focus on the images themselves (e_I), and (2) those that focus on the ingredients themselves (e_L), with the order of attention being reversed so that the former focus on the ingredients themselves (e_L), and the latter focus on the images themselves (e_I) [1].

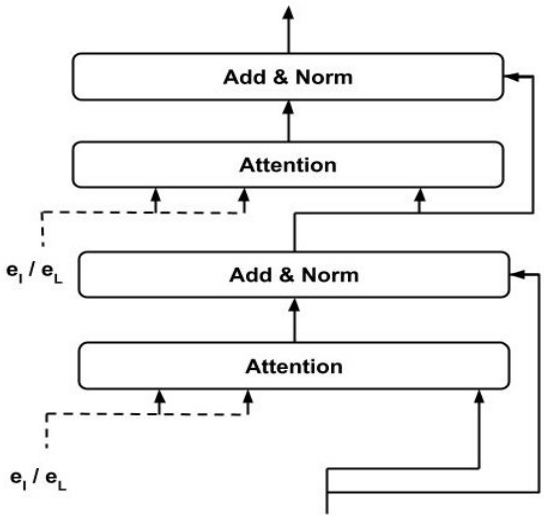

Figure 8: Sequential Attention

Independent focus produces the lowest outcomes, accompanied by both sequential and concurrent attention. Independent attention, on the other hand, is limited to a single level of refinement when it comes to achieving a desired result. For best outcomes, pay attention to concatenated attention as well. As a result, independent attention must contain information from both modes whereas concatenated attention might prefer one modality over the other. As a result, this study employs the concatenated attention approach to inform on the results of the test set.

6.4. OPTIMIZATION

Two steps are required to train the recipe transformer. In order to begin, we must pre-train the image encoder and the ingredients decoder. Next, we train the encoders and decoders by reducing the inverse log-likelihood and changing the coefficients of likelihood for each of the ingredients θ_R and the instructions θ_E, if appropriate.

7. MODEL EVALUATION

We assess models in this research using the Intersection over Union (IOU) and F1 scores, which are calculated for the aggregated values of True Positives (TP), False Negatives (FN), and False Positives (FP).

7.1. F1- SCORE

The F1-score is a metric that indicates the performance of the models on a given dataset. Using the harmonic mean of the system's precision and recall, it is possible to combine the system's precision and recall into a single value.

$$F1 = 2 \text{ x } \frac{precision \text{ x recall}}{precision + recall}$$

here,

$$precision = \frac{TP}{(TP+FP)}$$

$$recall = \frac{TP}{(TP+FN)}$$

From the above terms:

True Positives: Outcome in which the model forecasts the positive class properly.

False Positives: Outcome in which the model forecasts the positive class incorrectly.

False Negatives: Outcome in which the model forecasts the negative class incorrectly.

7.2. INTERSECTION OVER UNION

Intersection over Union is a performance measure that is used to determine an object detector's accuracy on a given dataset. To use Intersection over Union to analyse an object detector, it required bounding boxes for the ground truth (i.e., the original labelled bounding boxes from the dataset which specify the location of the object in the image) and the bounding boxes predicted by our model.

$$IOU = \frac{area\ of\ overlap}{area\ of\ union}$$

here,

$$area\ of\ overlap = A \cap B$$

$$area\ of\ union = A \cup B$$

A indicates the region of predicted bounded box

B indicates the region of original bounded box

The following table 1 depicts the accuracy of recipe generation (using IOU, F1-score) and table 2 shows the accuracy of ingredient prediction (using precision and recall).

Table 1: Accuracy of Recipe Generation

Model	IOU	F1
Recipe Generation	0.76	0.89

Table 2: Accuracy of Ingredient Prediction

Model	Precision	Recall
Ingredient Prediction	0.82	0.85

8. RESULTS AND DISCUSSIONS

Various convolutional neural network (CNN) models used for object recognition are compared to the suggested model in this research. The table 3 depicts the comparison of the model accuracies.

Table 3: Model Accuracy Comparison

Model	F1-score	Precision	Recall
Inception V3	0.47	0.45	0.50
VGG	0.54	0.52	0.58
ResNet-50	0.66	0.65	0.68
Our Model	0.81	0.80	0.83

Compared to other model, the proposed model outputs the highest accuracy when the terms like F1-score, precision and recall were considered.

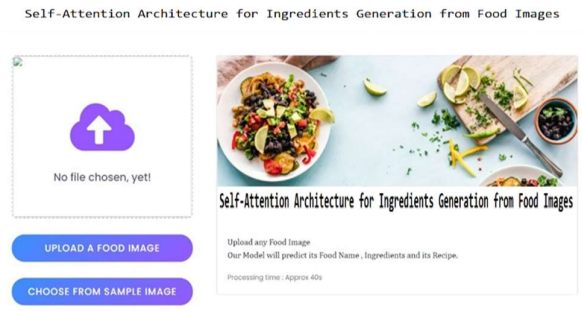

Figure 9: Home page of the web application

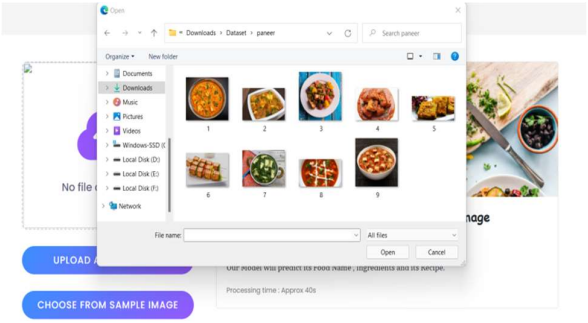

Figure 10: Select an image to predict

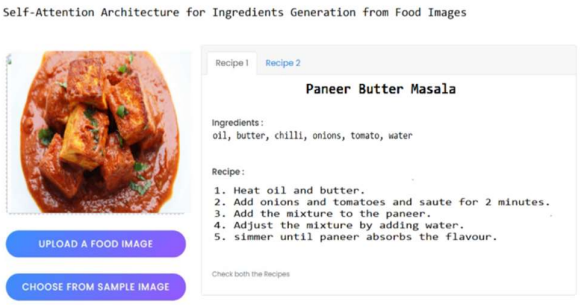

Figure 11: Predicting the title, ingredients and recipe for the image

CONCLUSION AND FUTURE SCOPE

An image-to-recipe creation system is shown in this article, which uses a food photograph to produce a recipe complete with a name, items, and a list of cooking directions. It first predicts ingredients from image, demonstrating the need of modelling dependencies. When it comes to creating instructions from images and inferred components, the model emphasises the need of considering both modalities at the same time. Users' study confirms the task's difficulty and establishes superiority over current image-retrieving technologies.

Additionally, a web application (figures 9, 10) is incorporated with the model, which aids in increasing the level of interaction between the user and the model. Title, ingredients list, and cooking instructions are all outputs (figure 11).

REFERENCES

[1]A. Salvador, M. Drozdzal, X. Giro-i-Nieto and A. Romero, "Inverse Cooking: Recipe Generation from Food Images," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 10445-10454, doi: 10.1109/CVPR.2019.01070.

[2]Micael Carvalho, Remi Cadene, David Picard, Laure Soulier,Nicolas Thome, and Matthieu Cord. Cross-modal retrieval inthe cooking context:Leaming semantic text-image embed-dings.In*SIGIR,*2018.

[3]J. Fujita, M. Sato and H. Nobuhara, "Model for Cooking Recipe Generation using Reinforcement Learning," 2021 IEEE 37th International Conference on Data Engineering Workshops (ICDEW), 2021, pp. 1-4, doi: 10.1109/ICDEW53142.2021.00007.

[4] Y. Pan, Q. Xu and Y. Li, "Food Recipe Alternation and Generation with Natural Language Processing Techniques," 2020 IEEE 36th International Conference on Data Engineering Workshops (ICDEW), 2020, pp. 94-97, doi: 10.1109/ICDEW49219.2020.000-1.

[5] A. Reusch, A. Weber, M. Thiele and W. Lehner, "RecipeGM: A Hierarchical Recipe Generation Model," 2021 IEEE 37th International Conference on Data Engineering Workshops (ICDEW), 2021, pp. 24-29, doi: 10.1109/ICDEW53142.2021.00012.

[6] W. A. d. Santos, J. R. Bezerra, L. F. Wanderley Góes and F. M. F. Ferreira, "Creative Culinary Recipe Generation

Based on Statistical Language Models," in IEEE Access, vol. 8, pp. 146263-146283, 2020, doi: 10.1109/ACCESS.2020.3013436.

[7]H. Jabeen, J. Weinz and J. Lehmann, "AutoChef: Automated Generation of Cooking Recipes," 2020 IEEE Congress on Evolutionary Computation (CEC), 2020, pp. 1-7, doi: 10.1109/CEC48606.2020.9185605.

[8] N. Hnoohom and S. Yuenyong, "Thai fast food image classification using deep learning," 2018 International ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (ECTI-NCON), 2018, pp. 116-119, doi: 10.1109/ECTI-NCON.2018.8378293.

[9] G. G. Lee, C. Huang, J. Chen, S. Chen and H. Chen, "AIFood: A Large-Scale Food Images Dataset for Ingredient Recognition," TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), 2019, pp. 802-805, doi: 10.1109/TENCON.2019.8929715.

[10]W. Xu, H. Sun, C. Deng and Y. Tan, "TextDream: Conditional Text Generation by Searching in the Semantic Space," 2018 IEEE Congress on Evolutionary Computation (CEC), 2018, pp. 1-6, doi: 10.1109/CEC.2018.8477776.

[11] K. Singla and S. Biswas, "Machine learning explanability method for the multi-label classification model," 2021 IEEE 15th International Conference on Semantic Computing (ICSC), 2021, pp. 337-340, doi: 10.1109/ICSC50631.2021.00063.

[12] B. Akhand and V. Susheela Devi, "Multi label classification of discrete data," 2013 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2013, pp. 1-5, doi: 10.1109/FUZZ-IEEE.2013.6622574.

[13]M. Huang and P. Zhao, "Image multi-label learning algorithm based on label correlation," 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE), 2021, pp. 606-609, doi: 10.1109/ICCECE51280.2021.9342484.

[14] Y. Li and Y. Wang, "A Multi-label Image Classification Algorithm Based on Attention Model," 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS), 2018, pp. 728-731, doi: 10.1109/ICIS.2018.8466472.

[15]B. Wang, Y. Liu, W. Xiao, Z. Xiong and M. Zhang, "Positive and negative max pooling for image classification," 2013 IEEE International Conference on Consumer Electronics (ICCE), 2013, pp. 278-279, doi: 10.1109/ICCE.2013.6486894.

[16] S. Yadav, Alpana and S. Chand, "Automated Food image Classification using Deep Learning approach," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), 2021, pp. 542-545, doi: 10.1109/ICACCS51430.2021.9441889.

[17]Weiqing Min, Bing-Kun Bao, Shuhuan Mei, Yaohui Zhu,Yong Rui, and Shuqiang Jiang.You are what you eat: Ex-ploring rich recipe information for cross-region food analy-sis.IEEETransactionsonMultimedia,2018.

[18] T. -H. Do, D. -D. -A. Nguyen, H. -Q. Dang, H. -N. Nguyen, P. -P. Pham and D. -T. Nguyen, "30VNFoods: A Dataset for Vietnamese Foods Recognition," 2021 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT), 2021, pp. 311-315, doi: 10.1109/COMNETSAT53002.2021.9530774.

[19] S. Mezgec and B. K. Seljak, "Using Deep Learning for Food and Beverage Image Recognition," 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 5149-5151, doi: 10.1109/BigData47090.2019.9006181.

[20]M. Kim, T. Kim and D. Kim, "Spatio-Temporal Slowfast Self-Attention Network for Action Recognition," 2020 IEEE International Conference on Image Processing (ICIP), 2020, pp. 2206-2210, doi: 991-995, doi: 10.1109/ICUS50048.2020.9274885.

[21] L. Wu, T. Tong, M. Du and Q. Gao, "Image Colorization Algorithm based on Self-Attention Network," 2020 Cross Strait Radio Science & Wireless Technology Conference (CSRSWTC), 2020, pp. 1-3, doi: 10.1109/CSRSWTC50769.2020.9372464.

[22] T. Kang and K. H. Lee, "Unsupervised Image-to-Image Translation with Self-Attention Networks," 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), 2020, pp. 102-108, doi: 10.1109/BigComp48618.2020.00-92.

[23] Y. Gao, H. Luo, W. Zhu, F. Ma, J. Zhao and K. Qin, "Self-Attention Underwater Image Enhancement by Data Augmentation," 2020 3rd International Conference on Unmanned Systems (ICUS), 2020, pp.

[24] Q. A. Al-Haija and A. Adebanjo, "Breast Cancer Diagnosis in Histopathological Images Using ResNet-50 Convolutional Neural Network," 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2020, pp. 1-7, doi: 10.1109/IEMTRONICS51293.2020.9216455.

[25] Q. A. Al-Haija and A. Adebanjo, "Breast Cancer Diagnosis in Histopathological Images Using ResNet-50 Convolutional Neural Network," 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 2020, pp. 1-7, doi: 10.1109/IEMTRONICS51293.2020.9216455.