M.Tech III Sem Project Report

Personalized Recipe Recommendation
A smart app

under the guidance of
Dr.T. Sridevi
Associate professor

Submitted By:
M.Samyuktha
(1601-19-742-016)

Department of Computer Science and Engineering
Chaitanya Bharathi Institute of Technology

Abstract

Recipes are a language of cooking. We understand how a certain dish can be made by reading a recipe. This project introduces a smart application to generate recipes using food image input. The model is trained on the Recipe 1M dataset and Google Recipe box API. The model has used Resnet neural networks to classify and label food images. We utilize the inverse cooking approach to recreate and generate new recipes given the classified food images. We predict the ingredients and generate the cooking instructions without imposing any order, by inferencing the image and different ingredients simultaneously. We propose a novel recipe generation application implementation to show (1) Performance improvement of ingredients prediction (2) Relevant recipe generation from the given input image with DenseNet neural network (3) Produce high quality compelling recipes which are better than information retrieval requiring human judgement.

Algorithm:

Step 1: Understanding the problem set and acquiring a relevant dataset

Given a set I = {ingredient 1, …, ingredient i , …, ingredient k} consisting of ingredients, recipe generation aims at generating a recipe for the user using ingredients available to them.

Example :

Sample ingredients 1 = {eggs, onion, bread}

Sample ingredients 2 = {Green tea, chocolate powder, baking soda, milk}

Step 2: Dataset

2.1 - Structure of dataset

Understanding the data structure of Recipe1M dataset which contains raw, highly unstructured data. The dataset can be grouped into 2 layers. The first layer consists of basic information free text - (title, list of ingredients, sequence of instructions for preparing the dish). The second layer contains any image (JPEG) associated with the recipe, annotations and builds upon the first layer.

2.2- Analysis of dataset:

Outliers exist for images: as several of the included recipe collections curate user-submitted images, popular recipes like chocolate chip cookies have orders of magnitude more images than the average. Notably, 25% of images are associated with 1% of recipes while half of all images belong to 10% of recipes; the size of the second layer in the number of unique recipes is 333k.

Recipe1M includes approximately 0.4% duplicate recipes and 2% duplicate images (different recipes may share the same image). Excluding those 0.4% recipes, 20% of recipes have non-unique titles but symmetrically differ by a median of 16 ingredients. 0.2% of recipes share the same ingredients but are relatively simple (e.g., spaghetti, granola), having a median of six ingredients.

Step 3: Learn Embedding

Neural joint embedding model builds upon recipe and image representation.

3.1. Representation of recipes

There are two major components of a recipe: its ingredients and cooking instructions.

Ingredients: The initial ingredient name extraction task is solved by a bi-directional LSTM that performs logistic regression on each word in the ingredient text. Training is performed on a subset of our training set for which we have the annotation for actual ingredient names. Ingredient name extraction module works with 99.5% accuracy tested on a held-out set.

Cooking Instructions: Each recipe has a list of cooking instructions. The instructions are quite lengthy (greater than 208 words). An LSTM model is designed to encode a sequence of a sentence and uses that encoding as context when decoding/predicting the previous and next sentences. The representation of a single instruction is the final output of the encoder.

3.2. Representation of food images

For the image representation, we adopt the Resnet-50 model. The deep residual networks take it to another level using ubiquitous identity mappings that enable the training of much deeper architectures with better performance.

Step 4: Joint Neural Embedding

The ingredients encoder is implemented using a bidirectional LSTM: at each time step it takes two-ingredient word2vec representations. The instructions encoder is implemented through a regular LSTM. At each time step, it receives an instruction representation from the skip-instructions encoder, and finally, it produces the fixed-length representation. The recipe and image representations are mapped into the joint embedding space.

Step 5: Semantic Regularization

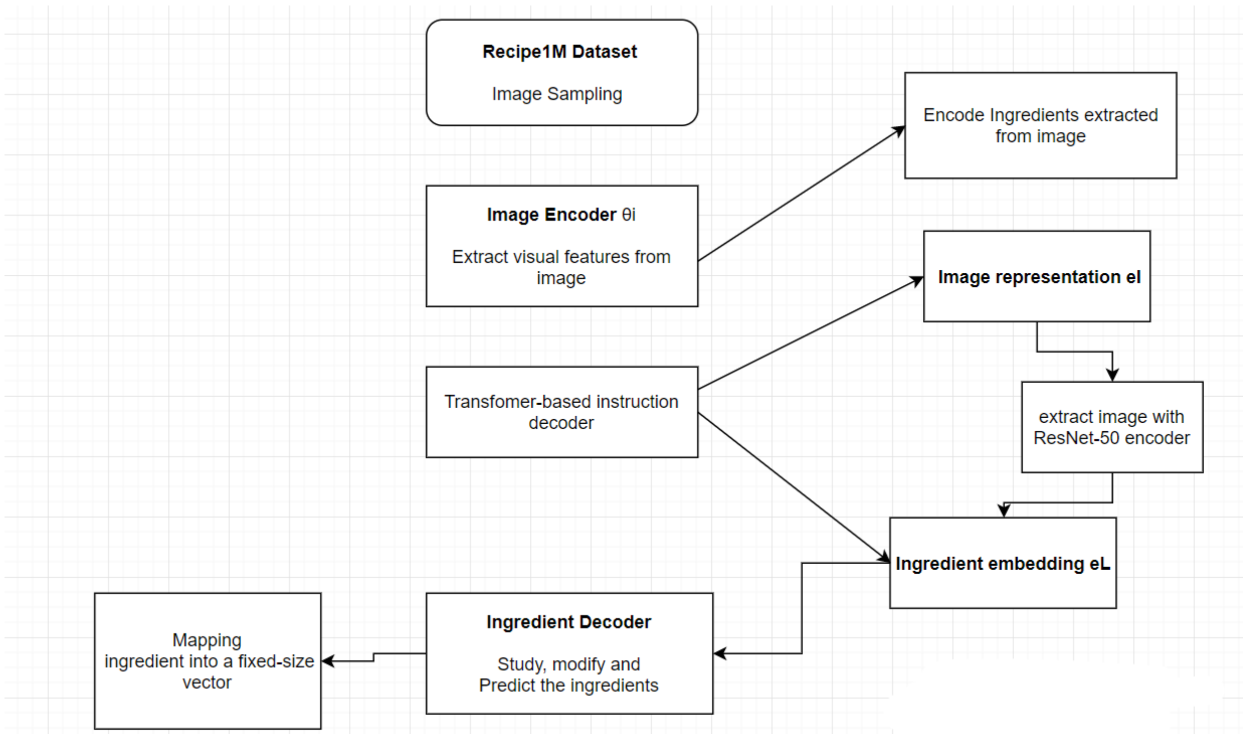
The model learns to classify any image or recipe embedding into one of the food-related semantic categories (recipe and image embeddings).

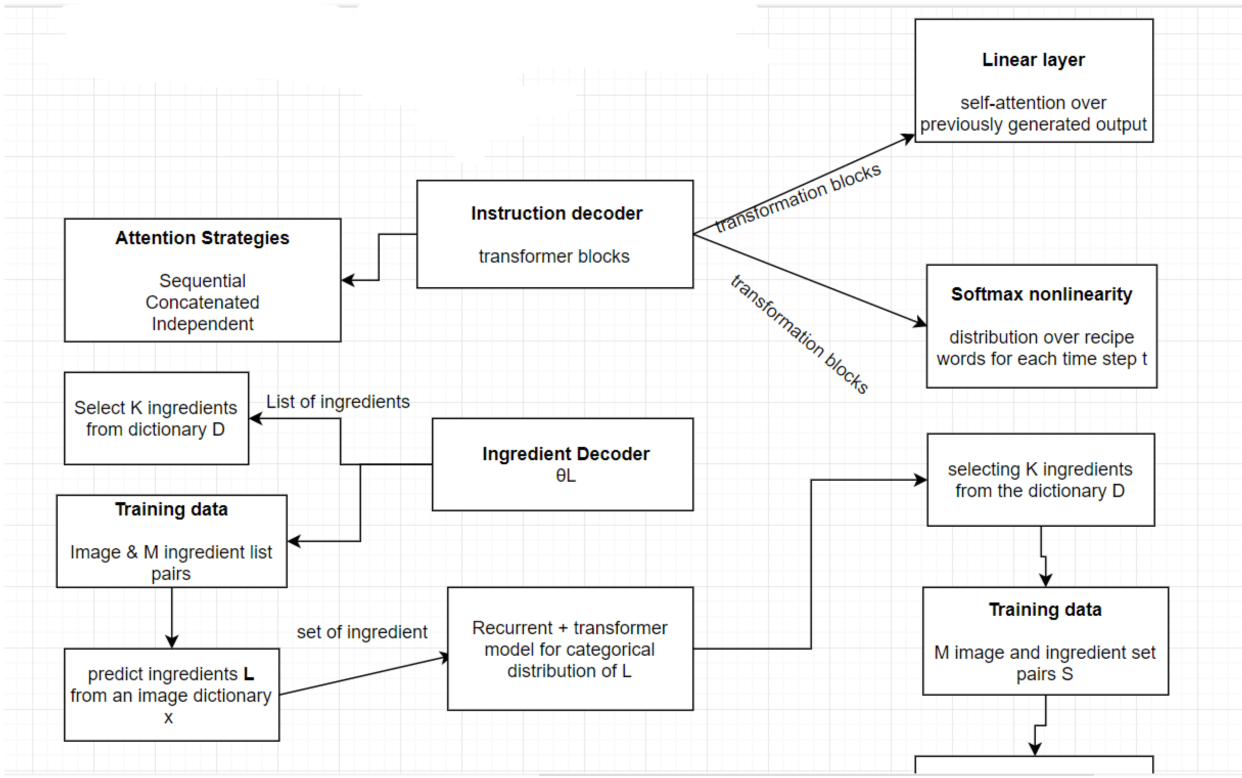
- Semantic Categories
- Classification
- Optimization

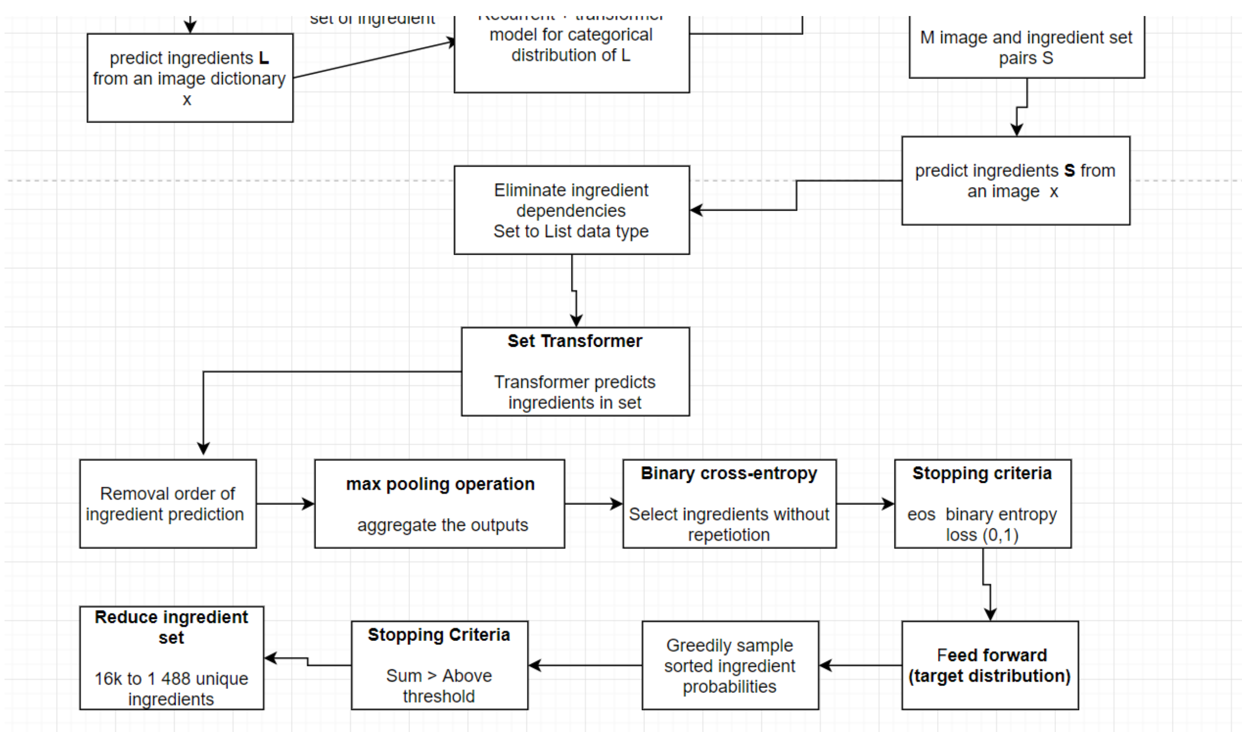
Step 6: Implement the Neural model with positive recipe-image pairs and a random negative recipe-image pair.

Flowchart

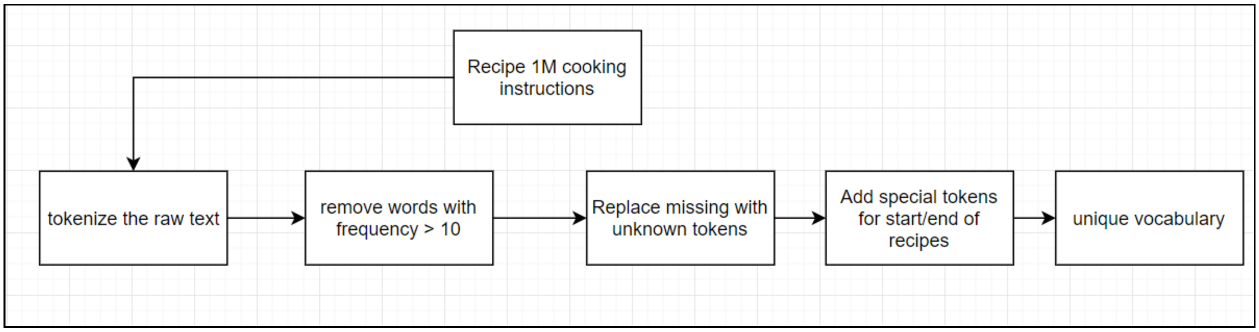
Flowchart for project process workflow



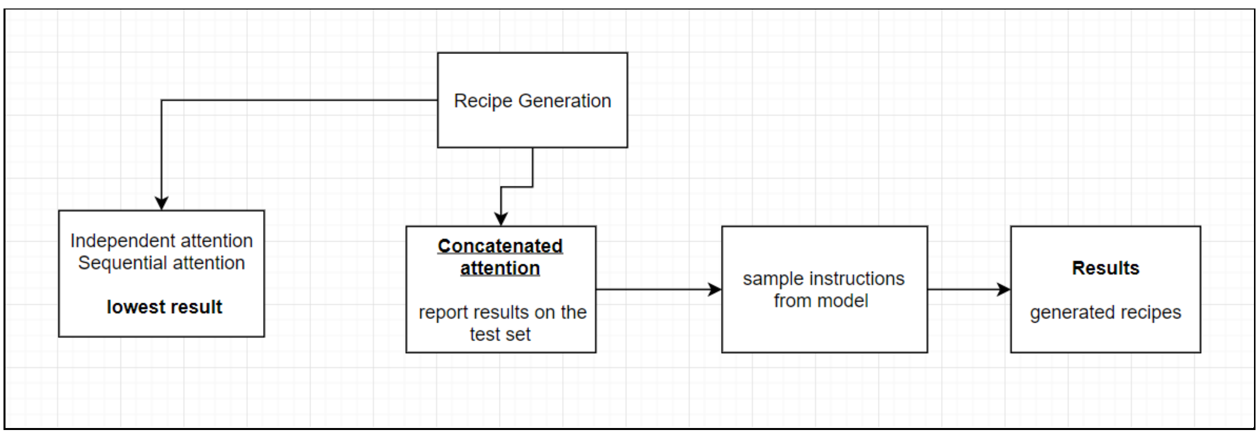

Linear layer
self-attention over previously generated output
transformation blocks
Instruction decoder
transformer blocks
Attention Strategies
Sequential
Concatenated
Independent
transformation blocks
Softmax nonlinearity
distribution over recipe words for each time step t
Select K ingredients from dictionary D
List of ingredients
Ingredient Decoder
θL
selecting K ingredients from the dictionary D
Training data
Image & M ingredient list pairs
Training data
M image and ingredient set pairs S
set of ingredient
Recurrent + transformer model for categorical distribution of L
predict ingredients **L** from an image dictionary x


model for categorical distribution of L
M image and ingredient set pairs S
predict ingredients **L** from an image dictionary x
predict ingredients **S** from an image x
Eliminate ingredient dependencies
Set to List data type
Set Transformer
Transformer predicts ingredients in set
Removal order of ingredient prediction
max pooling operation
aggregate the outputs
Binary cross-entropy
Select ingredients without repetition
Stopping criteria
eos binary entropy loss (0,1)
Reduce ingredient set
16k to 1 488 unique ingredients
Stopping Criteria
Sum > Above threshold
Greedily sample sorted ingredient probabilities
Feed forward (target distribution)

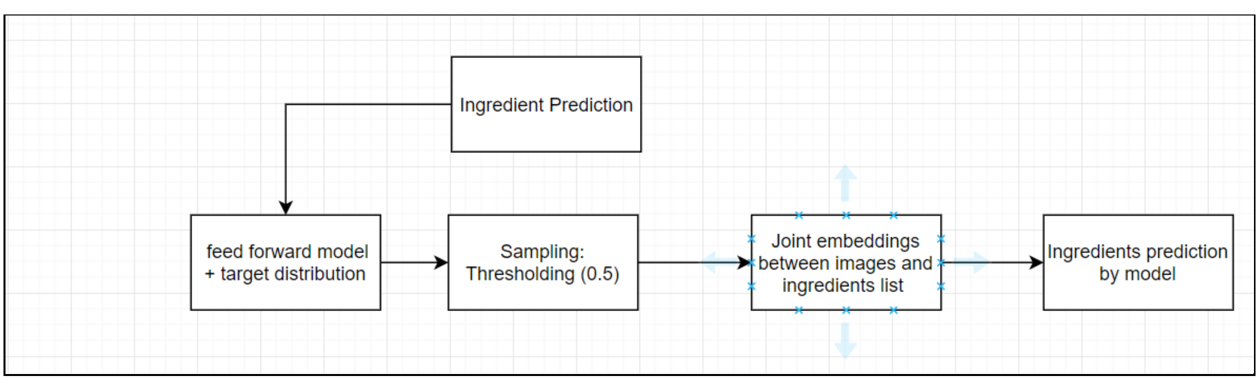
Flowchart for Data Preprocessing - Cooking Instructions



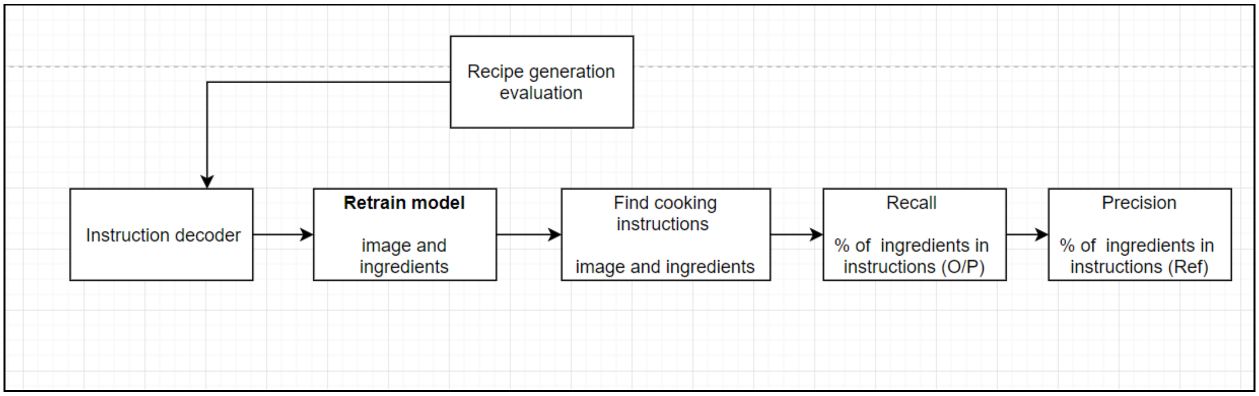
Flowchart for Recipe Generation process



Flowchart for the Ingredient Prediction process



Flowchart for Recipe generation evaluation process



Dataset

Given the relevance of understanding recipes, it is surprising that there is not a larger body of work on the topic. We estimate that this is due to the absence of a large, general collection of recipe data. The recipes are however raw HTML.

Although the aforementioned datasets constitute a large step towards learning richer recipe representations, they are still limited in either generality or size. As the ability to learn effective representations is largely a function of the quantity and quality of the available data, In comparison to the current largest dataset in this domain, Recipe1M includes twice as many recipes and eight times as many images.

The recipes were scraped from over two dozen popular cooking websites and processed through a pipeline that extracted relevant text from the raw HTML, downloaded linked images, and assembled the data into a compact JSON schema in which each datum was uniquely identified. As part of the extraction process, excessive whitespace, HTML entities, and non-ASCII characters were removed from the recipe text.

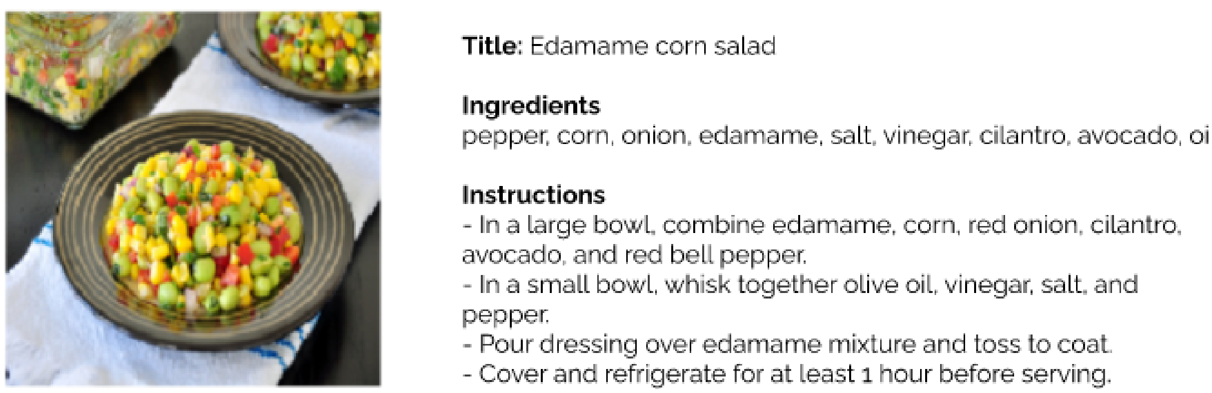
The contents of the Recipe1M dataset may logically be grouped into two layers. The first contains basic information including the title, a list of ingredients, and a sequence of instructions for preparing the dish; all of these data are provided as free text. The second layer builds upon the first and includes any images with which the recipe is associated–these are provided as RGB in JPEG format. Additionally, a subset of recipes is annotated with course labels (e.g., appetizer, side dish, dessert).

The Recipe1M dataset is composed of 1 029 720 recipes scraped from cooking websites.Since the dataset was obtained by scraping cooking websites, the resulting recipes are highly unstructured and contain frequently redundant or very narrowly defined cooking ingredients (e.g. olive oil, virgin olive oil and spanish olive oil are separate ingredients).

Expected Result

Prediction of ingredients and generate the cooking instructions without imposing any order, by inference the image and different ingredients simultaneously.

Qualitative Results



Collaboration with companies - None

Base paper reference

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), Long Beach, CA, USA, 2019, pp. 10445-10454, doi: 10.1109/CVPR.2019.01070.