NeuralCook – Image2Ingredients and Cooking Recommendation using Deep Learning

Sumith Reddi Baddam
Indiana University Bloomington
srbaddam@iu.edu

Sai Swapna Gollapudi
Indiana University Bloomington
sagoll@iu.edu

Sanath Keerthi Edupuganti
Indiana University Bloomington
saedup@iu.edu

Abstract

In this paper, we present a deep learning application to extract ingredients of photographed cooking dishes and provide appropriate recipes and ingredients. The application further recommends dishes to cook, given a set of ingredients from the inventory. Parsing an image of a dish and identifying its ingredients is not a complex task for humans. Indeed, people can quickly identify the dish and its ingredients just by looking at it. But it is much more complex for computers. To produce systems that can achieve this, we combine current state-of-the-art techniques in both Computer Vision and Natural Language Processing to learn the semantic knowledge from both images and recipes using joint embeddings. We propose three deep learning models to improve the performance of the system. These models classify the image of the dish, predict the ingredients present in the image, and fetch similar dish images. The model also recommends the dishes that can be cooked from a set of ingredients.

1. Introduction

In this fast-paced life, it is hard to find the time to cook, especially if you are feeding a family. Sure, baking for other people can be fun, but cooking meals can feel more like a chore. People are always looking for ways to make it easier to fit cooking into their busy schedules.

Foodstagramming is so popular because people believe they are interesting and that, of course, that they get a lot of joy from beautiful, tasty food. It turns out that cooking, especially when done for others, comes with some positive psychological benefits [1]. Posting aesthetic snapshots of gourmet dishes is also a part of our visual self-presentation: polished photos show us in a positive light to others

Since food photos are huge on social media and are easy-to-produce and relevant to everyone, we built a deep learning model to identify the ingredients present in a food image and provide the recipe for people to cook.

For instance, when browsing through your feed you see a dish cooked by your friend and get cravings, just download the image, and upload it to the **NeuralCook** model. The model runs the neural network algorithms and identifies what ingredients are present in the dish and gives you a recipe.

Furthermore, if you have certain ingredients and not sure what to cook or need suggestions, you can query the ingredients to the NeuralCook model and get the recommended recipes and dishes.

2. Background and related work

Food item classification and recommendation of the ingredients as well as recipes started with the boom in the datasets available from the past 6-7 years. Bossard et al. [2] introduced the Food-101 visual classification dataset and set a baseline of 50.8% accuracy. Although the dataset used is similar dataset this model just classifies the images.

Newer and fine-grained approaches emerged like Facebook AI research and a team at MIT has built an application pic2recipe which takes an image as an input and retrieves the recipe for this image from its database. Amaia Salvador [3] proposed a cross-modal embedding on the Recipe1M dataset where the modal outputs a recipe for a given input image and vice-versa. Herranz et al. [4] proposed a multi-model framework integrating content (food image), context (geolocation), and external data like similar food images, recipes, restaurants at the location and nutritional data.

Chang Liu [5] proposed a new CNN model for deriving food information like food type and portion size from food image but this application does not recommend any other images of the dish from its database. Facebook Inverse cooking recipe generation application also generates recipes directly from images using a retrieval-based approach. In this paper we present a new dataset with better attention to detail on the ingredients for the Food-101 dataset and propose a pipeline for Image classification and Ingredient recommendation.

Given an Image our model classifies the food item in the image and gives ingredients required to prepare that dish and vice-versa. Along with that we also provide images of food items in the dataset that match the query image, or the ingredients inputted.

3. Methodology

The motivation for building this application is to provide people with recipes for dishes given the image and to suggest the dishes they can cook with the ingredients they have. Parsing an image of a dish and identifying its ingredients is not a complex task for humans. Indeed, people can quickly identify the dish and its ingredients just by looking at it. But it is much more complex for computers. To produce systems that can achieve this, we need to combine current state-of-the-art techniques in both Computer Vision and Natural Language Processing.

Fig1: Data Architecture flow diagram

3.1. Dataset

The distinction between the difficulty of the chosen problem and previous supervised classification problems is that there are large overlaps in food dishes, as dishes of different categories may look very similar only in terms of image information.

To address this complexity, we used the datasets from multiple sources. Food-101, allrecipes.com and Recipe1M+ are the primary sources. Our dataset consists of over 120,000 images and 5000 ingredient types. Food-101 dataset consists of images of food, organized by the type of food into 101 categories. It is first used in the paper, "Food-101 – Mining Discriminative Components with Random Forests" by Lukas Bossard, Matthieu Guillaumin and Luc Van Gool [6]. For the ingredients, we web scraped allrecipes.com using python scripts and regular expressions.

3.2. Data Pre-processing

The ingredients data obtained from web scrapping consists of unstructured texts. For instance, ingredients are texts like "1 tablespoon of salt", "3/4 cup of rice". Here, tablespoon and cup represent the quantity and not ingredient. Also, few ingredient texts consisted of similar same ingredients with different names, like "bbq sauce" and "barbeque sauce", "smith apples" and "granny smith apples".

To clean such data and to extract ingredient names from ingredients texts, we used NLP operations, text analysis, pre-processing, and keywords identification. To cluster the similar ingredients into one bucket, we used and open source software, OpenRefine, which clusters the words together based on various distance metrics and algorithms.

The images from food-101 and allrecipes.com are preprocessed, cleaned, and resized into 512 x 512.

3.3. Training

The diagram below depicts the architecture of the two deep learning models, which we use to learn the joint embeddings for the images and ingredients. We utilize the paired (recipe and image) data to learn a common embedding space.

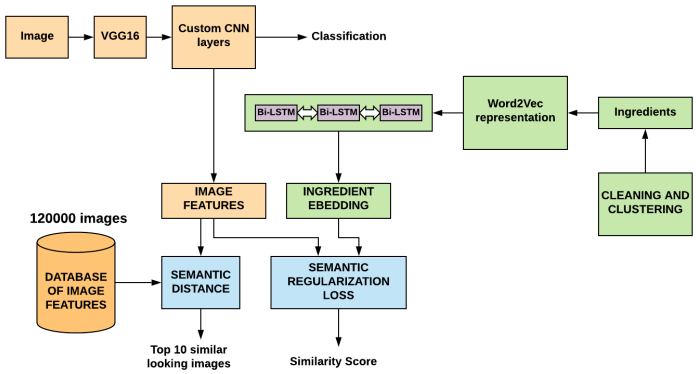

Fig2: Flow diagram for Training Joint Embedding

We train three deep learning models to learn the joint embeddings for the images and ingredients/recipe. Here we utilize the paired (recipe and image) data to learn a common embedding space.

First, for the image representation we adopt state-of-the-art 16-layer VGGNet pre-trained on ImageNet as a base network and build a customized **Convolutional Neural Network** classifier on top of it to categorize the input dish image into one of the 101 categories.

Fig3: Training sample images, french fries, sushi, tiramisu, baklava, apple pie and fried calamari, respectively.

We then remove the softmax classification layer from the model and use the output from the last fully connected layer to encode the images into 4096-dimension embeddings.

Secondly, before building the ingredient model, we clean and pre-process the ingredient text. For each ingredient we learn an ingredient level embedding representation. To do so, we train a customized **Bidirectional LSTM (Bi-LSTM)** to learn the features and the embedding space like that of the first model. The rationale for using a Bi-LSTM is that the ingredients list is an unordered set.

Finally, the ingredients embeddings and image embeddings are used to learn the **Joint Embeddings**. In simple terms, the goal is to learn transformations to make the embeddings for an image-ingredient pair close. To ensure this closeness, the model is trained with **cosine similarity loss**.

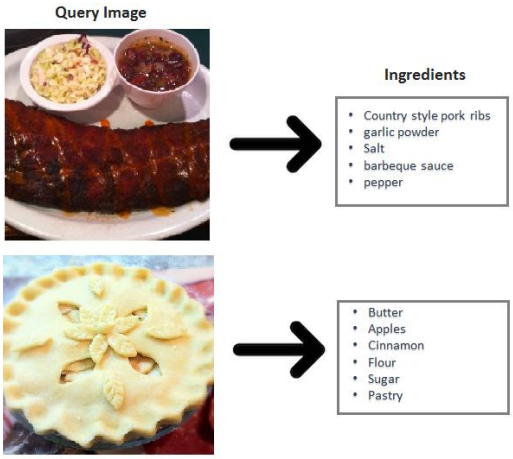

Fig4: Query image and ingredient predictions by model

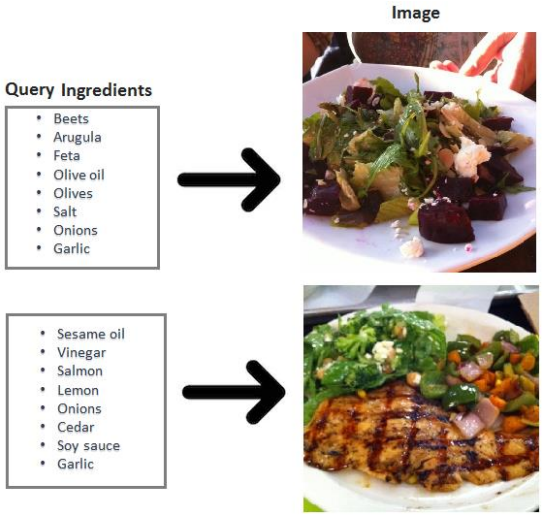

Fig5: Query ingredients and output dish recommendations

3.4. Testing using web application

We built web application and consumable REST APIs for users to integrate them into their applications, evaluate and capture feedback. The architecture for the application is depicted below.

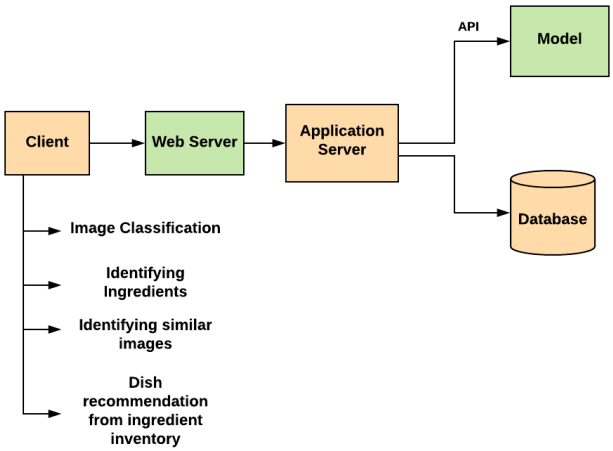

Fig6: Web application design and architecture

The image shows a 5-layered application with client, web server, application server, modeling, and database. The trained and saved models are run using python scripts and the web applications are built using Node JS.

The functionalities of the application include, classification of an image, predicting the ingredients in the dish's image, fetching similar-looking images, and recommending the dishes given a set of ingredients. The client interacts with the web server by making HTTP get or put requests.

4. Results

We evaluated the application with 10,000 images from various categories. The food image classification, ingredient embedding, and recommendation retrieval systems are evaluated separately and as a whole for a better understanding of the model performance. We also built REST API for each of the three models so that the application will be open source and consumable to other users.

4.1. Image classification

The input image is first passed through the CNN network for classification. The classification model when run on 10,000 images, achieved 85% accuracy.

Fig7: Classification model output

4.2. Ingredient retrieval from image

Fig8: Ingredients prediction from images

As you can see from the image, the second dish I uploaded was chicken fried rice, which we cooked at home. Since the model was not trained on chicken fried rice category, it assigned the dish to the closest one, which is, fried rice. This is a universal problem. There is no way any model can categorize every dish that is present.

This is where the joint embedding and language model we built helps us identify the ingredients and recipe. We see that the Bi-LSTM identifies chicken breasts as one of the ingredients, which is what was missing from the image analysis.

4.3. Similar images retrieval

The similar images retrieval model is evaluated based on the top 10 and top 5 recommendations. The top-10 model achieved an accuracy of 95% and top-5 achieved 91%.

This similarity model is used to prune the results further, the application goes a step further to provide similar images to the user from the 120,000 images database.

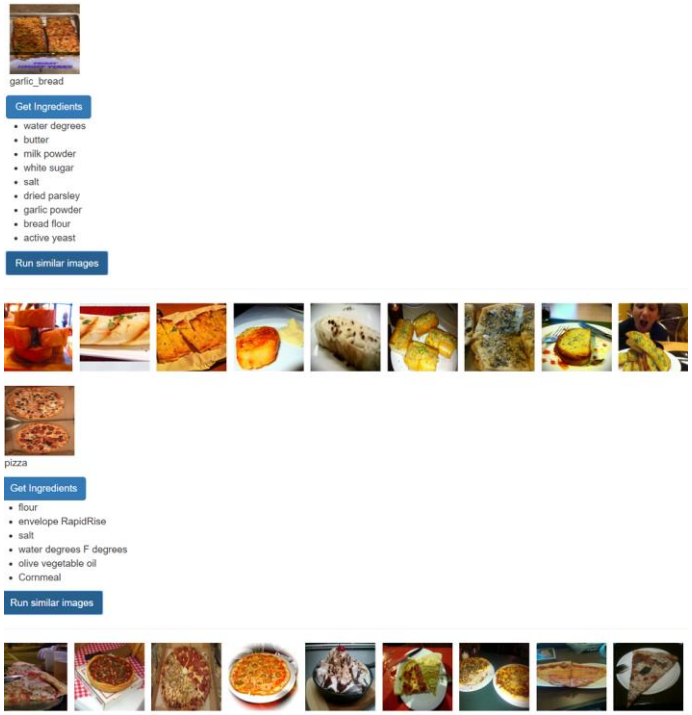

Fig9: Similarity images retrieval model outputs

4.4. Dish recommendation from ingredients inventory

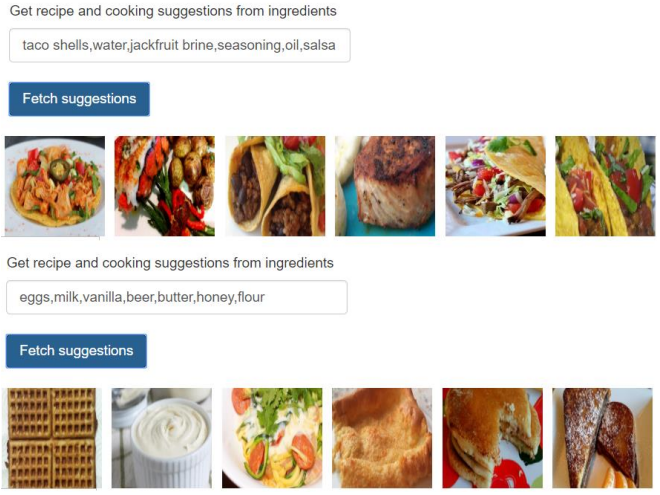

Fig10: Recommendation of dishes given ingredients

5. Discussion

The above results show that our model is giving promising results for each of the 4 tasks: Image classification, Ingredient retrieval, Similar images retrieval, dish recommendations from ingredients inventory. We have tested on 10000 images and achieved 85% accuracy for classification, the top 10 similar images gave an accuracy of about 95% and the top 5 similar images gave an accuracy of about 91% manually tested on 100 ingredients which are 90% accurate. However, the model does not always give perfect results. This could be because the model is limited to only 101 food categories, the poor quality of the input image, image taken in a different perspective, multiple dishes in a single image, ground truth is different from the actual food item in the image or missing ingredients in the training data which could be because of preprocessing or due to anomalies in data.

Fig11: Output of classification and similar images for taco

In the above example, a taco image was classified as macaroni and cheese by the model and for similar images the model has given images of macaroni and cheese. This could be because of the input image, which is focusing more on the stuffing in the taco which looks like mac and cheese.

Fig12: Ingredients prediction for a fried rice image

In the above example: the model has correctly classified the image as fried rice however the main ingredient which is rice is missing in the ingredients. This is due to the missing ingredients in training data.

6. Conclusion

Considering the increase in the amount and the popularity of food images in social media which might interest the users to try out different dishes they see. Seldom people know the ingredients they need to prepare the dish, or they might not know the name of the dish they see. We address this in this paper, by presenting The Neural Cook module to identify dishes, extract ingredients, and suggest similar looking dishes. Further the model also recommends dishes given a set of ingredients. We present this idea in the form of an interactive web application that is open-sourced and makes it easier for the users to get the ingredients and names of their favorite foods. We plan on extending this model further to fetch recipes from a given image and fetch images given a recipe as well as to extend the scope of the number of categories.

7. References

[1] The Very Real Psychological Benefits of Cooking For Other People: https://www.huffpost.com/entry/benefits-of-cooking-for-others_n_5967858ae4b0a0c6f1e67a15

[2] L. Bossard, M. Guillaumin, and L. Van Gool. Food-101–mining discriminative components with random forests. In European Conference on Computer Vision, pages 446–461. Springer, 2014.

[3] Learning cross-modal embeddings for cooking recipes and food images, Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, Antonio Torralba; The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 3020-3028.

[4] L. Herranz, W. Min, and S. Jiang, Food recognition and recipe analysis: integrating visual content, context and external knowledge,CoRR, vol. abs/1801.07239, 2018. [Online]. Available: http://arxiv.org/abs/1801.07239

[5] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, and Y. Ma, Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment, in International Conference on Smart Homes and Health Telematics. Springer, 2016, pp. 3748

[6] The Food-101 data set consists of images from Foodspotting which are not the property of the Federal Institute of Technology Zurich (ETHZ)

[7] All Recipes—https://www.allrecipes.com/