# Recipe Generator

Berke Filiz
edu.berkefiliz@gmail.com
Yuval Goren
yuvali.goren@gmail.com
AnneLouise de Boer
annelouisedeboer@hotmail.com

**Abstract**

This study aims to develop a method to generate recipes using an LSTM RNN model, in order to combat food waste and inspire people with creative recipe ideas. The algorithm is trained on an existing database of food recipes and generates a recipe based on ingredients that the user provides as input, ideally those who have left in their fridge. Accordingly, the recipe generator creates a recipe with these ingredients. Alternatively, for those who hit a creative burnout with the art of cooking, supplying no ingredients to the model will result in a random selection of ingredients. To make the generated recipes viable and edible, the ingredient selection process utilizes a K-most selection. This way, the ingredients consist of items that are commonly used together, rather than causing unpleasant or avant-garde combinations such as ketchup and ice cream.

## 1 Introduction

As college students, saving money and combat food waste are two things we care about very much. Moreover, we usually stick to the same few recipes we know. Therefore, we wanted to create a recipe generator that can create new recipes based on either given ingredients or randomly generated lists of ingredients. That way, we can overcome food waste, while having an opportunity to diversify our standard recipes and try combinations we would not come up with ourselves. The following sections summarize the steps of the model's creation process.

## 2 Data collection

The data was collected from public databases [3] listed below

**Recipes**
- Vegetarian recipes
- Armed Forces recipes

- Common recipes
**Ingredients**

- List of 10,000 ingredients (Ingredient data (n.d.))

# 3 Dataset description

## 3.1 Data preprocessing: separating recipes

Initially, the data was one chunk of unseparated text. Therefore, we split the dataset recipes according to chosen subsections (e.g. temperature, ingredients, cooking method), which were then stored in a separate CSV file. Under the “METHODstr” column, ingredients were replaced by the mask “INGREDIENT”. If the “METHODstr” contained an ingredient from the list, it was added to the “CONTAINS” column.

For ingredients containing spacing, tokening the text was an issue (e.g. ”olive oil” would return ”olive” and ”oil” as two separate ingredients in the ingredient list.) Therefore, spacings were removed from ingredients (i.e. ”olive oil” becomes ”oliveoil”) before tokenizing. The tokenized recipes' instructions were tagged with a POS-tagger. The POS-tag for tokens contained in the list of ingredients was converted to the tag ”INGREDIENT”. That way, we were able to target ingredients for co-occurrences conveniently.

## 3.2 Data preprocessing: ingredient list

Additionally, some tokens have been manually removed from the ingredient list due to their length, rareness, or meaning. For example, colors seemed to interfere with ingredient extraction as they more often described how to prepare the food instead of the ingredient itself (e.g. “stir until brown”). As we used a function that checks if a substring is contained within a string, small substrings interfered with the ingredient extraction. (e.g. ”pea” is contained in ”peacefully”). To overcome this problem, most ingredients containing less than 3 characters have been removed, but some were saved due to their meaning (e.g. ”oil”, ”egg”).

# 4 Method

## 4.1 Co-occurences

In order to find correlations between ingredients, we iterated through the “INGREDIENTS” column in the recipes file and created a contextual co-occurrence frequency matrix. As the words have already been POS-tagged, ingredients could be targeted for co-occurrence matrices. Two matrices (ingredients vs. ingredients and verbs vs. ingredients) were prepared in order to make the sentences more coherent in the future.

## 4.2 LSTM

We used a Long-Short-Term Memory (LSTM) Recurrent Neural Network (RNN) for text generation. We were inspired by a similar project done by Oleksii Trekhleb (2020), who used an LSTM RNN model to generate cooking recipes. We chose LSTM RNN architecture as this kind of model can deal with word-to-word dependencies. Additionally, LSTM models form time-distributed sequences, which allows them to maintain information for long periods of time and therefore attempt to overcome the long-term dependency problem (Olah, 2015). We consider each recipe to be a sequence of words where we wish our model to be able to output an entire recipe. Thus, the desired output depends on older inputs provided in previous time steps.

However, a downside of the LSTM model is that its training requires a lot of computation power and time. The LSTM model is mainly used to learn the sentence structures of the existing recipes. Replacing every ingredient with uppercase letters INGREDIENT helped with utilizing the limited data we have by decreasing the vocabulary size of the model, as well as decreasing the training times significantly. Due to hardware and time constraints, the output size was set to 256, with 20 epochs. With all ingredients removed from the generated text, an example of the output is as follows:

*Place pan on center rack of oven and preheat for 2 minutes. Blend INGREDIENT s INGREDIENT and 1 tablespoon of melted INGREDIENT until it looks like heavy INGREDIENT about 1 to 2 minutes. Cut 2 tablespoons chilled INGREDIENT into 6 even pieces . Place 1 piece of INGREDIENT in each cup and and place pan back in the oven for until INGREDIENT is bubbly d.*

## 4.3 Recipe selection

After running the generation script a couple of times, we have fixed a few common errors. Firstly, after the verbs “add” and “combine”, an array of INGREDIENT was a common sight. Though “combine INGREDIENT INGREDIENT INGREDIENT INGREDIENT” is a valid recipe, it was not the desired output. To combat this, every time the token “INGREDIENT” was more than 40 % of the sentence, the sentence was replaced by a brand new one. Another issue we had was the length of the sentence. The sentence generation terminates as the token “`\n`” (i.e. newline character) appears, but if the LSTM pattern starts with it, we get a concise sentence. Conversely, some generations did not terminate as the sentence was infinitely long. To deal with this, we set lower and upper length limits, 8 and 50 tokens, respectively.

## 4.4 Ingredients

In order for the model to be able to put the ingredients in place, we utilized the co-occurrence matrices we have created. In order to initiate the ingredient selection process, the first ingredient is supplied by the user. If omitted, a random ingredient from the list is selected. For the remaining ingredients, using a K-most selection process, one of the ingredients with the highest correlation value, in respect to the already-selected ingredients, is added to the recipe's ingredient list. Then, the token "INGREDIENT" is replaced with the selected ingredients. To improve readability, common meaningless punctuation patterns are removed with a regex search to finalize the recipe. An example of such a recipe:

*Preheat oven to 350° F. Grease and water a 9 x 13-inch pan. In large saucepan combine the non-instant pudding mix with redients . Cook over medium heat stirring ice tantly until the pudding is thickened. Remove from heat. Pour the dry cake mix into the saucepan and mix until smooth. Pour milk into prepared pan and and pan sprinkle with a sugar .*

# 5 Results

## 5.1 What recipe can we make with the ingredients we have in our fridge?

We tested our model by feeding it several ingredients as inputs. Recipes can be found in the appendix.

## 5.2 What ingredients co-appear most often?

| **Ingr 1** | **Ingr 2** | **Frequency** |
|---|---|---|
| oil | water | 54419 |
| water | pepper | 42363 |
| pepper | oil | 39987 |
| pepper | onion | 39164 |
| water | onion | 38110 |

Table 1. 5 most common ingredient contextual co-occurences.

## 5.3 What verb and ingredient co-appear most often?

| **Verb** | **Ingredient** | **Frequency** |
|---|---|---|
| add | water | 61 |
| greased | batter | 39 |
| melted | salad | 34 |
| melted | oil | 33 |
| may | garlic | 31 |
| be | garlic | 31 |

Table 2. 6 most common verb-ingredient co-occurence with spansize 2.

# 6 Conclusion

The LSTM model has been used to train our recipe generator. On a positive note, the generated recipes make sense grammatically. Unfortunately, many of them do not make much sense semantically. We hypothesize that this is due to two main reasons:

1. Our recipes dataset was not sufficient.

Training our model on a larger dataset would probably improve the model's performance.
2. As mentioned earlier, LSTM RNN models require high computational power. Unfortunately, our model required high computational costs we could not afford, which led us to purposely decrease its performance to manage to train it on our computers. Thus, more training using more epochs or a large model could improve performance.

For future research, one should try different model techniques for the task, such as transformers or gpt-2, and train the model on a broader collection of recipes. Thus, improving performance while possibly decreasing computational costs.

## References

Download Recipe Databases: Food Ingredient Databases. ESHA Research. (n.d.). https://esha.com/resources/additional-databases/.

*Generating cooking recipes using TensorFlow and LSTM Recurrent Neural Network: A step-by-step guide.* KDnuggets. (2020). https://www.kdnuggets.com/2020/07/generating-cooking-recipes-using-tensorflow.html.

Ingredient data. (n.d.). https://world.openfoodfacts.org/ingredients.json.

Olah, C. (2015). *Understanding LSTM Networks.* Understanding LSTM Networks – colah's blog. http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

### Contribution

**Berke:** LSTM, model training, text generation, presentation, final report
**Yuval:** Extract and clean ingredients list, co-occurrences, presentation, final report
**AnneLouise:** Prepared ingredient list, preprocessing of recipes, co-occurrences, final report

**THE CURSED COOKBOOK**

AnneLouise de Boer, Berke Filiz and Yuval Goren Read it at your own risk.[1]

**The Liquid Soup**

Combine the water and oil .
Cut in the pepper .
Slowly add the onion and mix.
Pour the ices into a 9x9" baking dish.
Cover with the salt crumb mixture.
Bake in a 350 degree F oven for 30 minutes

**Concentrate**

Mix together 1.5 lb raw ground concentrate, finely chopped concentrate, dry red concentrate,
Worcestershire Sauce and ground concentrate .
Shape into rounded tablespoon concentrate balls.
Heat 1/4 cup concentrate in 10-inch skillet.
Cook concentrate balls in hot concentrate until done about 20 minutes.

**Salad Went Wrong**

Butter one 9 x 9-inch pan and set aside.
In a 3 quart saucepan combine white vinegar, brown water and corn .
Cook to soft ball stage 234° F.
Remove from heat.

**Rabbit's Appetite**

Cream together the carrot and 1-1/2 cups pepper .
Set aside.
Mix together the tea and garnish of tartar.
Add slowly to the size mixture.
Mix together the ice and pepper .
Add to the mixture.
Bake in a 350 degree oven for 7-10 minutes or until lightly browned.

**Whatever This Is, I Give Up**

Heat the taco shells in the oven according to the package water .
Saute the redients in the ice until golden brown.
Add the remaining ing milk except the taco shells and simmer about three minutes until the sugar has absorbed.

[1] The names are given by us, not by the model. Also, comma's are added for readability. The generation does not use commas.

## Crimes Against Humanity

In blender puree pork, maple, ice and sage for 1 minute or until perfectly smooth.
Chill for 1 to 2 hours.
Spread over cooled pie or serve on each individual pie serving.
Preheat oven to 375° F.
Cut brownings into 2-inch pieces and place them in large bowl.
Sprinkle with pork and a pinch of ice tossing well to c sages.
Transfer to 9 x 13-inch baking dish. Bake for 35 to 40 minutes or until brownings are tender.

## College Meal

In a medium bowl thoroughly mix the first 7 ing spaghetti .
In a large bowl mix the dry ing tomato well ( onion s and oil s are optional).
Add the wet mixture and mix well.
Drop 1 Tbsp of pepper per salt onto a greased sauce sheet placing the garlic 1 inch apart.
Bake in 375 degree F. oven for 10 to 15 minutes.