Remy Final Report

Bob Li

Wendy Wang

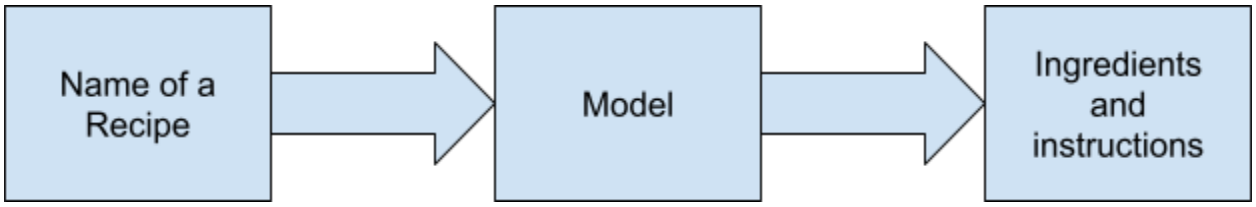
Word count: 1913

	permission to post video	permission to post source code	permission to post final report
Bob	Yes	Yes	Yes
Wendy	Yes	Yes	Yes

Introduction

The goal of our project is to create a recipe generator using a lightweight transformer model. The input of the model will be the name of a recipe and the output are the ingredients and instructions for the recipe. We are motivated by our passion for food and drinks and want to know if we can generate creative new recipes for food and drinks without the need for culinary expertise. Since transformer models are already good at generating a variety of text, we believe they can be trained to generate recipes as well.

Illustration of project goal



Background & Related Work

There are two papers that stand out in the field of recipe generation. The first called RecipeNLG by Bień et al. aims to improve the quality of recipe generation by improving the dataset of recipes. It builds upon Recipes 1M+ which is a dataset that contains both images and text descriptions of over one million recipes [1]. They showed that improving the dataset by data cleaning and deduplication resulted in better quality of generation without any changes to the model and used several metrics including BLEU to measure the performance of their model [1]. The second paper by Katserelis et al. focuses on generating recipes specifically in the fine-dining category. They scraped recipes and fine-tuned a GPT-2 model [2]. Both papers used special tokens to impose the structure of the dataset (e.g., a token for each the title section, ingredient section, and instruction section).

We aim to extend on their work by experimenting with a much smaller and selective dataset to try to understand how specific training data can affect the quality of

generated recipes. We also hope to discover a threshold of samples needed to generate reasonable recipes.

Data and Data Processing

For our dataset, we chose to focus on two types of recipes: cocktails and cakes. We collected 302 cocktail recipes from liquor.com and 385 cake recipes from allrecipes.com. We used 292 cocktail recipes and 375 cake recipes in our training dataset and saved the remaining 10 of each for evaluation.

We created our dataset by collecting URLs from the respective websites and created a custom HTML parsing script to retrieve the recipe name, ingredients, and instructions. For cocktails, we cleaned the dataset by removing batch drink recipes such as punch bowls and kept only single serving recipes. We also removed any instructions to create special ingredients in order to keep the ingredients atomic. For cakes, we removed any brand name ingredients, sorted the ingredient list, and removed content in brackets. We then formatted the dataset by adding by adding headers for each part of the recipe: <CAKE/DRINK> RECIPE NAME, <CAKE/DRINK> RECIPE INGREDIENTS, <CAKE/DRINK> RECIPE INSTRUCTIONS. Samples of cake and drink recipes are found in their respective links.

For training, the recipes were truncated to 1024 tokens to fit the model and padded at the end with end of sequence (EOS) tokens.

Architecture and Software

We used a pretrained GPT-2 model from HuggingFace by calling

```
AutoModelForCausalLM.from_pretrained("gpt2")
```

to load the pre-trained model. The model has 12 layers, 768 hidden states, 12 heads and a total of 117M parameters.

We created three different fine-tuned models by changing the dataset such that

- Model A: Fine-tuned on cocktails only
- Model B: Fine-tuned on cake only
- Model C: Fine-tuned on both cocktails and cake

The models were fine-tuned to predict the next word in the recipes using the following hyperparameters:

- Batch size = 2,
- Learning rate = 2e-5
- Weight decay = 0.01
- Epochs = 50

We also used the OpenAI playground to experiment with GPT-3 using the prompt “write a <cake/cocktail> recipe called <the “invented” name of your recipe>”.

Quantitative Results

We had 10 cocktail recipes and 10 cake recipes reserved for evaluation. To evaluate our models, we used two methods to evaluate our results. The first method BLEU is commonly used to compare a generated result to a reference. BLEU is scored on a range of 0 to 1 where 1 represents the highest similarity to the reference. We also created a set of guidelines to evaluate the quality of a recipe based on the ingredients and instructions.

The ingredients were given a score out of 5 based on the following criteria:

- Ingredients are reasonable and in the correct quantities
    - Cocktails generally include 1-2oz of alcohol and some flavouring
    - Cakes generally include ingredients such as flour, sugar, eggs, butter
- Does not include repeated ingredients
- Ingredients relate to the name of the recipe
    - This can be ignored for weird recipe names such as “Red Hook”
    - This should not be ignored when obvious ingredient hints are included such as “Mandarin Orange Cake”

The instructions were also given a score out of 5 based on the following criteria:

- Instructions make use of all the listed ingredients
- Instructions do not include ingredients which were not in the ingredients list
- Instructions make sense
    - E.g. You can garnish a drink with mint, but you can’t garnish a drink with an egg

- Instructions are easy to follow

For each recipe in the evaluation set, we generated 5 completions using the recipe name as a prompt with the appropriate header for cocktail or cake. We then compared these generated recipes to the original recipe using BLEU. The BLEU scores were average and the results are summarized in table 1. The best BLEU result was then given a recipe quality score using the ingredient and instruction scoring guidelines above. Since there are 10 recipes for each category, the total recipe quality score was out of 100 and shown in table 2.

Table 1: Average BLEU scores to evaluate similarity to reference

BLEU Range [0, 1]	Model trained on single dataset	Model trained on both datasets
Cocktails	0.19 (A)	0.19 (C)
Cake	0.15 (B)	0.16 (C)

Table 2: Evaluation of recipe quality based on guidelines

Recipe Quality Score /100	Model trained on single dataset	Model trained on both datasets
Cocktails	93 (A)	87 (C)
Cake	72 (B)	77 (C)

We can see from the low BLEU scores that the recipes generated were not close to the reference recipes. However, the recipe quality scores show us that the recipes were decent in quality and reasonable to follow. The recipe quality score for cocktails was lower on model C compared to model A due to some confusion between cakes and

cocktails. Both BLEU and the recipe quality scores show that cake recipes improved when the model was trained on the combined cakes and cocktails dataset.

Through experimentation, we found that GPT-3 is able to generate existing recipes easily. To level the playing ground, we came up with 4 cocktail names and 4 cake names for fictional recipes. These are shown in table 3. We generated one completion per prompt and evaluated the recipe quality, summarized in table 4.

Table 3: Recipe names used to compare model C and GPT-3

Cake	Tooth Fairy	Beyond Meat	Hogwarts	Red Wedding
Cocktail	Tooth Fairy	Berlin Night	Bullet Train	Wave Rider

Table 4: Comparison of recipe quality for novel recipes

Recipe Quality Score /40	Model C	text-davinci-003
Cocktails	30.5	38.5
Cake	30.5	40

We can see from these results that GPT-3 outperforms model C in terms of recipe quality in both categories.

Qualitative Results

We observed that model A is less creative than model C in generating drink recipes. 5/10 of the recipes generated by model A had liqueur + lemon juice + simple syrup whereas only 2/10 recipes generated by model C followed this format. An example of such recipes generated by model A is found here. For model B, we noticed that nuts and cinnamon were commonly included in recipes with 8/10 nut recipes and 6/10 cinnamon recipes. In contrast, 7/10 recipes from model C contain pecans/walnuts, and only 1/10 contains cinnamon. It appears that the models trained on single category

datasets tended to generate the average recipe more often whereas training on both datasets improved the probability of generating different recipes.

We also found that model C was better at connecting the recipe ingredients to the recipe name when possible. In one example shown here, model C was able to include orange juice when prompted for a recipe for “Mandarin Orange Cake”. In contrast, model B’s generated a recipe with pineapple instead of orange. In total, model C had 4/10 recipes with ingredients related to the name while model B only had 1/10.

Though the quality and creativity of some recipes generated by model C were better than A and B, model C also experienced some confusion between cocktails and cake. When prompted for a cocktail recipe called “Girls Next Door”, it generated a recipe with a mix of alcohol and cake ingredients such as butter and sugar which is shown here. The model could have been confused by the inclusion of cream of coconut which sounds like it could be used in a cake.

GPT-3 on the other hand is able to include arbitrary themes in the generated recipes. When prompted for a cake named “Beyond Meat”, GPT-3 was able to generate an innovative recipe. An excerpt of the recipe is shown below.

```
[...] 3. Place the Beyond Meat patties in a food processor and
pulse until they are broken down into small pieces. [...] 8.
Fold in the Beyond Meat pieces and chopped nuts, if desired.
[...]
```

For a cake named “Hogwarts”, GPT-3 was also aware of house colors from the Harry Potter Series.

```
[...]  7. To decorate, spread a thin layer of buttercream
frosting over the cooled cake. Sprinkle with red, yellow, and
green sprinkles to represent the colors of the Hogwarts house
flags. Enjoy! [...]
```

Discussion and Learnings

From our results, we believe our model is performing decently, but could be improved by more training data. It was surprising that training on both cakes and cocktails improved the quality and creativity of recipes. We could use this finding to improve the model by training on a wider variety of recipes. We anticipated that the model may be confused between cakes and cocktails so we tried adding CAKE and DRINK to the headers, however this was not enough and model C still experienced

some confusion. To get less confusion, we could try training a classifier to help filter the generated results.

We were unsurprised that GPT-3 outperformed our models given that it is much larger and trained on much more data. This doesn't mean however that smaller models can't achieve comparable performance when trained on the right data. Our method of manually collecting URLs was very tedious and we should have looked into automation methods in order to collect a larger, but still highly specialized, dataset.

If we were to start another similar project, we would try to include more information in training data and the prompt. Recipe names are often not very descriptive of what is in the recipe so a prompt describing the recipe such as "sweet and spicy cocktail" might yield better results.

Individual Contributions

Wendy

- Created html parsing scripts for allrecipes.com and liquor.com
- Cleaned up data
- Processed dataset for input to GPT-2 model using HuggingFace API
- Fine-tuned GPT-2 on combined dataset
- Evaluated and compared generated recipe quality from Models A, B, C

Bob

- Gathered URLs for dataset
- Fine tuned GPT-2 models on cocktail dataset and cake dataset
- Generated all prompts and completions from GPT-2 and GPT-3 for evaluation
- Evaluated GPT-2 models using BLEU
- Evaluated and compared generated recipe quality from GPT-2 and GPT-3

Reference

[1] M. Bień, M. Gilski, M. Maciejewska, W. Taisner, D. Wisniewski, and A. Lawrynowicz, "Recipenlg: A cooking recipes dataset for semi-structured text generation," ACL Anthology. [Online]. Available: https://aclanthology.org/2020.inlg-1.4/.

[2] K. Katserelis and K. Skianis, "Towards fine-dining recipe generation with generative pre-trained transformers," arXiv.org, 26-Sep-2022. [Online]. Available: https://arxiv.org/abs/2209.12774.