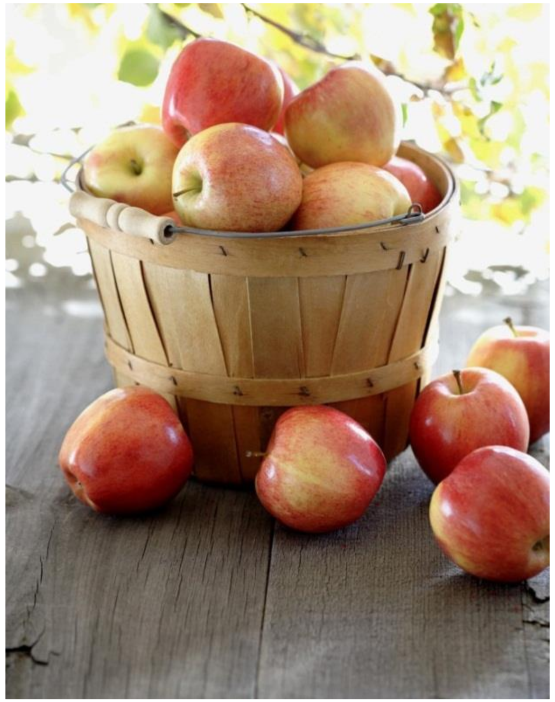
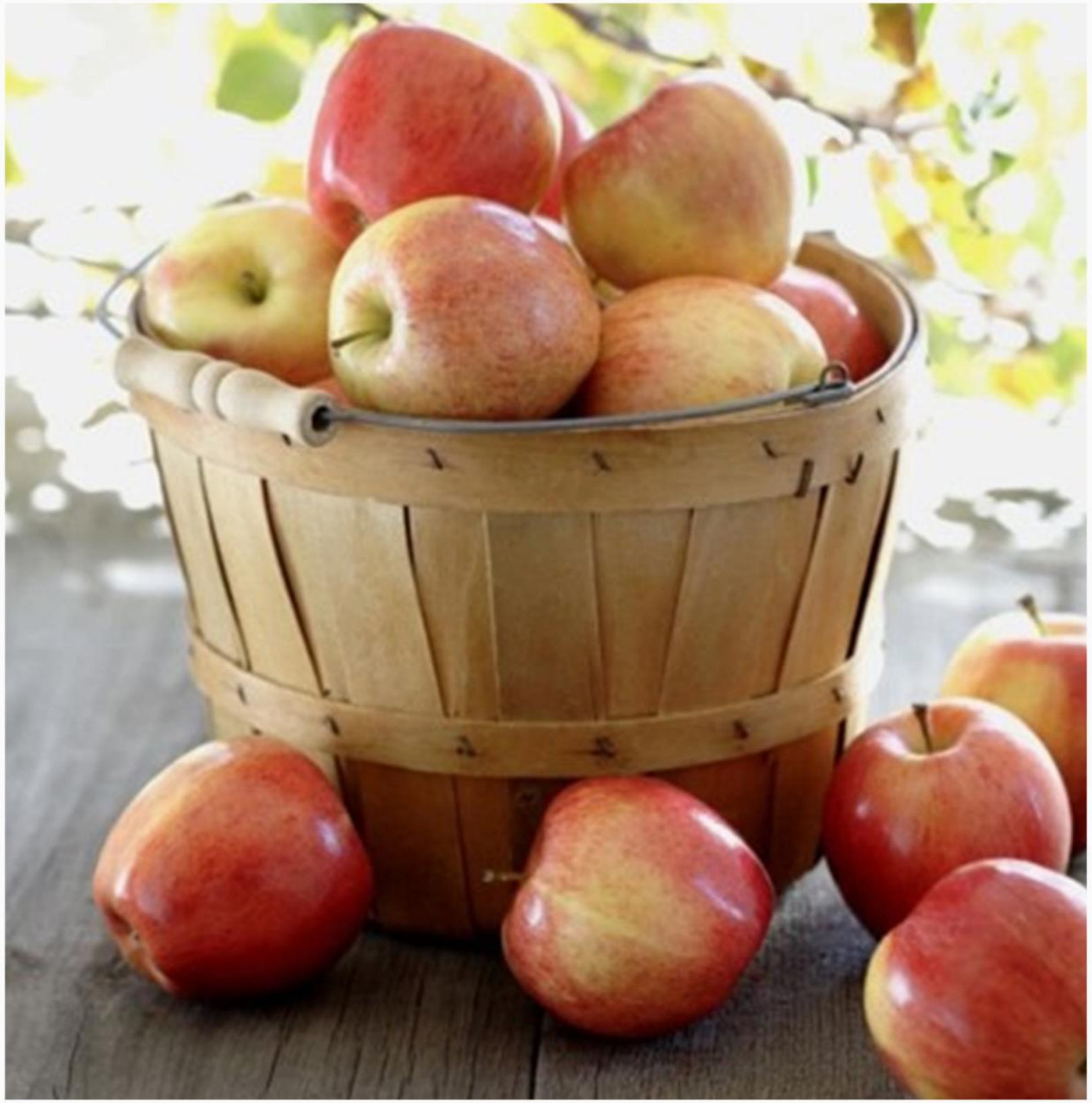


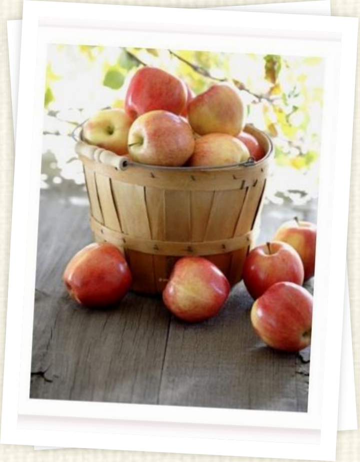
Food Recommender Based on Taste

Ranggajaya Ciptawan

Contents

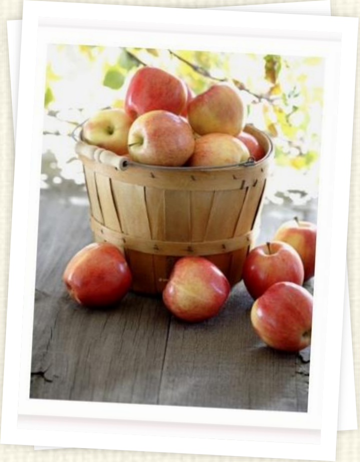


Introduction



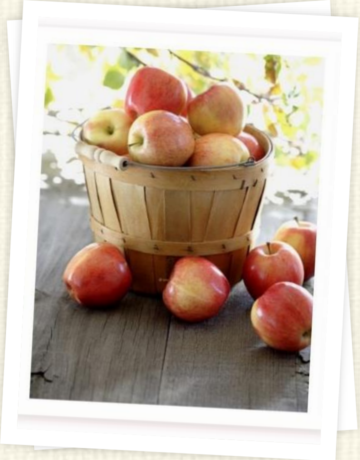
Introduction

- Currently, there are so many applications that can show you many choices of food.
- You can also filter the food based on its type (ex. Snacks, Chinese Food, Western Food, etc.).
- Furthermore, you can ask the restaurant to deliver it to your currently location!



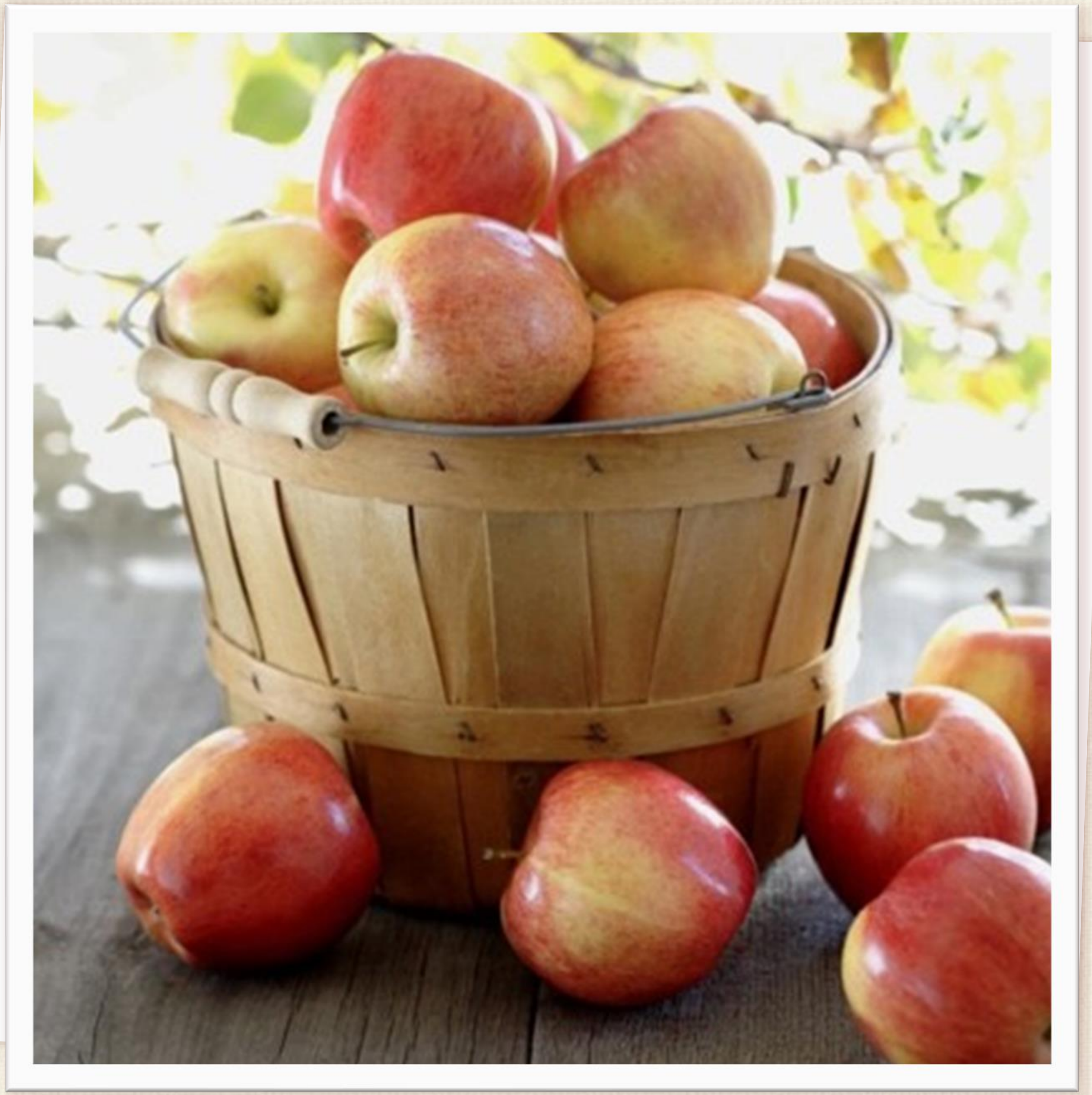
Introduction

- Sometimes, have you ever felt that you want to eat something but you don’t know what to eat specifically?
- Something like “I want to eat spicy food!” or “I want to eat something sweet!”.
- Until now, I haven’t found any application that can provide such a thing.



Introduction

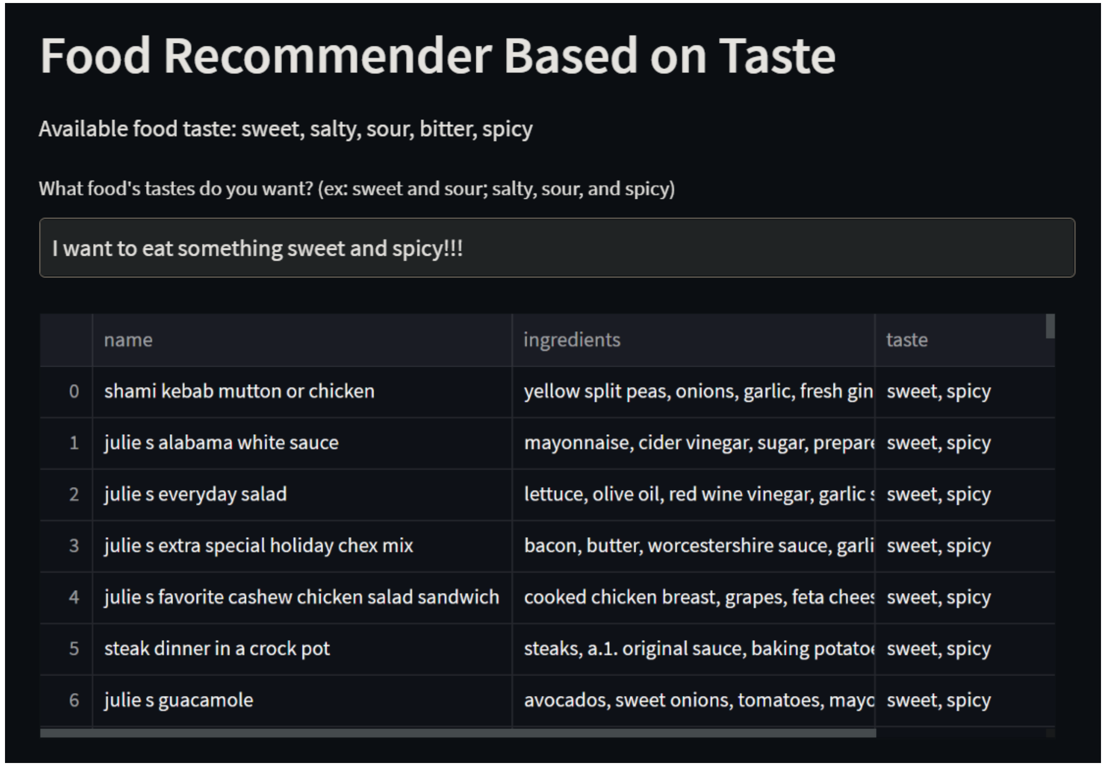
- Since I'm also a food lover, this motivates me to create something that can give me food recommendations based on my currently taste preference.
- So in this slide, I'd like to share what I made and also how I made it.
- I'm going to explain it a little bit detail so that you can understand what I made and then you can give me any feedback to improve it since I'm also still learning it.
- But don't worry! I'm going to explain it in a fun way. ☺



What did I do?

What did I do?

- So, what did I do?
- Well, like I said earlier, I tried to make a food recommender based on user’s currently taste preference.
- You can see the example on this figure on the right.



Food Recommender Based on Taste

Available food taste: sweet, salty, sour, bitter, spicy

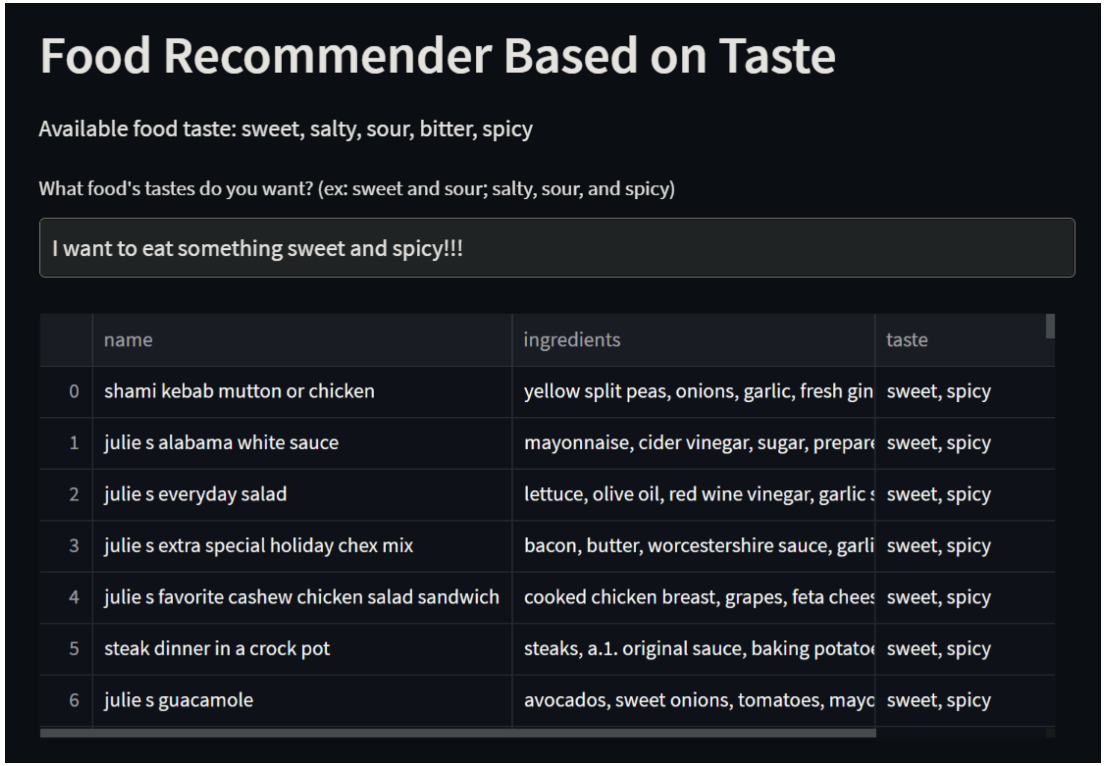
What food's tastes do you want? (ex: sweet and sour; salty, sour, and spicy)

I want to eat something sweet and spicy!!!

	name	ingredients	taste
0	shami kebab mutton or chicken	yellow split peas, onions, garlic, fresh gin	sweet, spicy
1	julie s alabama white sauce	mayonnaise, cider vinegar, sugar, prepare	sweet, spicy
2	julie s everyday salad	lettuce, olive oil, red wine vinegar, garlic s	sweet, spicy
3	julie s extra special holiday chex mix	bacon, butter, worcestershire sauce, garli	sweet, spicy
4	julie s favorite cashew chicken salad sandwich	cooked chicken breast, grapes, feta chees	sweet, spicy
5	steak dinner in a crock pot	steaks, a.1. original sauce, baking potato	sweet, spicy
6	julie s guacamole	avocados, sweet onions, tomatoes, mayo	sweet, spicy

What did I do?

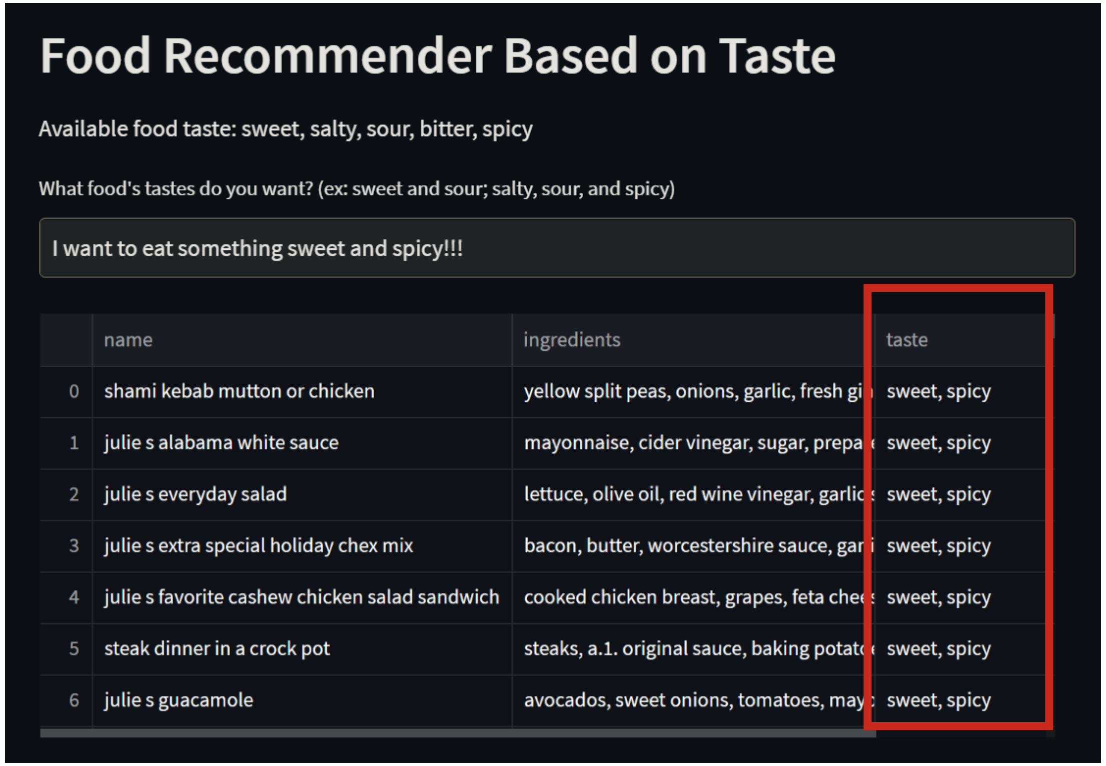
- As you can see, you can tell the system what kind of food taste you want to eat.
- Then, the system will give you some food recommendations along with its ingredients.
- It's very simple, isn't it? XD



	name	ingredients	taste
0	shami kebab mutton or chicken	yellow split peas, onions, garlic, fresh gin	sweet, spicy
1	julie s alabama white sauce	mayonnaise, cider vinegar, sugar, prepare	sweet, spicy
2	julie s everyday salad	lettuce, olive oil, red wine vinegar, garlic s	sweet, spicy
3	julie s extra special holiday chex mix	bacon, butter, worcestershire sauce, garli	sweet, spicy
4	julie s favorite cashew chicken salad sandwich	cooked chicken breast, grapes, feta chees	sweet, spicy
5	steak dinner in a crock pot	steaks, a.1. original sauce, baking potatoe	sweet, spicy
6	julie s guacamole	avocados, sweet onions, tomatoes, mayo	sweet, spicy

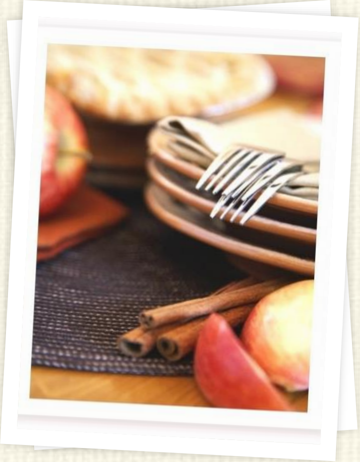
What did I do?

- The problem is, the dataset I got didn't provide the food's taste (I'll explain about the dataset on the next section).
- So, the most challenging part is "how can I classify the food taste without any deep knowledge about culinary? I'm not a chef! I don't even know how to boil water!"
- We'll answer that question later. ☺



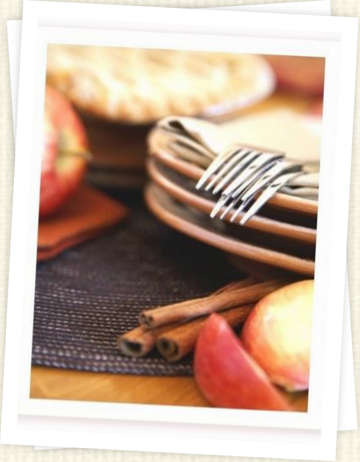
	name	ingredients	taste
0	shami kebab mutton or chicken	yellow split peas, onions, garlic, fresh gi	sweet, spicy
1	julie s alabama white sauce	mayonnaise, cider vinegar, sugar, prepa	sweet, spicy
2	julie s everyday salad	lettuce, olive oil, red wine vinegar, garlic	sweet, spicy
3	julie s extra special holiday chex mix	bacon, butter, worcestershire sauce, gar	sweet, spicy
4	julie s favorite cashew chicken salad sandwich	cooked chicken breast, grapes, feta chee	sweet, spicy
5	steak dinner in a crock pot	steaks, a.1. original sauce, baking potato	sweet, spicy
6	julie s guacamole	avocados, sweet onions, tomatoes, may	sweet, spicy

What dataset did I use?



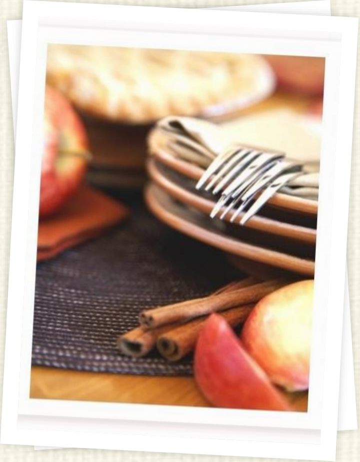
What dataset did I use?

- I used a dataset found on Kaggle. Here is the link.
- This dataset is crawled from Food.com.
- Food.com is a website that provides huge amount of food recipes. People can add their recipes and other people can comment and rate the recipes.
- Actually, the foods are already classified into several types (ex. Breakfast, Brunch, Healthy Food, etc.), but not its taste.
- It's a good website so I recommend you to visit it. ☺



What dataset did I use?

- Did I use all of the data? Of course not! I only used **Raw_recipes.csv**.
- Did I use all of the columns? Of course not! I only used two columns, **name** and **ingredients**. I wanted to use **steps,** but the data became too big to be saved on Github. So just leave it be.
- **name**: food's name.
- **ingredients**: food's ingredients.
- Here is the sample looks of the dataset.



What dataset did I use?

name	ingredients
arriba baked winter squash mexican style	['winter squash', 'mexican seasoning', 'mixed spice', 'honey', 'butter', 'olive oil', 'salt']
a bit different breakfast pizza	['prepared pizza crust', 'sausage patty', 'eggs', 'milk', 'salt and pepper', 'cheese']
apple a day milk shake	['milk', 'vanilla ice cream', 'frozen apple juice concentrate', 'apple']
…	…

How did I classify the food taste?

How did I classify the food taste?

- Now this is what we’ve been waiting for.
- After seeing the raw dataset, you must have wondered “how do we know if this food is sweet or spicy? ”.
- The most obvious answer is of course by checking the ingredients! That’s why I chose that column.
- I’m sure you will answer “yeah I know. But how precisely?”.

How did I classify the food taste?

- There were three possible ways that crossed my mind before.
- First, asking the food expert, such as chef, food nutritionist, etc. Then, we can classify the food taste manually or automatically. That's what we call *expert system*.
- Second, we can depend on our instinct to classify the food taste. For example, the "*a bit different breakfast pizza*" in the dataset sample before might be salty and spicy since it uses salt, pepper, and cheese.
- Third, we can use *semi-supervised learning** and this is what I used. ☺
- *notes: I don't know if this method is categorize as *semi-supervised learning.* You can verify it after reading my method.

How did I classify the food taste?

- So, the idea is, we will classify some of the foods first, and then training a model with those data to classify the other foods.
- One of the dumbest way to classify the food is by its title. XD
- For example, "*windy's **sweet** and **sour** meatballs*" will be classified as **sweet** and **sour** food.
- Let's just assume that the food's taste is guaranteed by its title. ☺

How did I classify the food taste?

- We will express the food taste with a binary vector.
- For example, sweet and sour will be expressed as [1 0 1 0 0], salty and spicy will be expressed as [0 1 0 0 1], etc.
- Always remember the order! [sweet, salty, sour, bitter, spicy].
- The *get_taste* function will do the job. It will be implemented to all rows.

```
def get_taste(x):
    tastes = ["sweet", "salty", "sour", "bitter", "spicy"]
    taste_result = []

    for taste in tastes:
        if taste in x:
            taste_result.append(1)
        else:
            taste_result.append(0)

    return taste_result

used_data["taste"] = used_data["name"].map(get_taste)
used_data.head()
```

How did I classify the food taste?

- To make it more readable, we will convert the vector into the taste's name.
- The result will look like the bottom right figure.

```
def get_taste_name(x):
    tastes = ["sweet", "salty", "sour", "bitter", "spicy"]
    taste_result = []

    for i in range(5):
        if x[i] == 1:
            taste_result.append(tastes[i])

    return str(taste_result)

used_data["taste_name"] = used_data["taste"].map(get_taste_name)
used_data.head()
```

	name	ingredients	taste	taste_name
88	souper easy sweet sour meatballs	['ground beef', 'dried breadcrumbs', 'onion', ...	[1, 0, 1, 0, 0]	['sweet', 'sour']
89	sour cream avocado dip vegan	['soft silken tofu', 'avocado', 'chunky salsa'...	[0, 0, 1, 0, 0]	['sour']
90	spicy banana bread	['shortening', 'white sugar', 'bananas', 'eggs...	[0, 0, 0, 0, 1]	['spicy']
276	slightly spicy black bean burgers	['black beans', 'oil', 'frozen corn', 'garlic'...	[0, 0, 0, 0, 1]	['spicy']
297	windy s sweet and sour meatballs	['lean ground beef', 'eggs', 'salt', 'black pe...	[1, 0, 1, 0, 0]	['sweet', 'sour']

How did I classify the food taste?

- Are we finished? Of course not!
- As you can see on the right figure, some of the foods are not yet labeled.
- That's because not all foods are titled with its taste.

	name	ingredients	taste	taste_name
0	arriba baked winter squash mexican style	['winter squash', 'mexican seasoning', 'mixed ...	[0, 0, 0, 0, 0]	[]
1	a bit different breakfast pizza	['prepared pizza crust', 'sausage patty', 'egg...	[0, 0, 0, 0, 0]	[]
2	all in the kitchen chili	['ground beef', 'yellow onions', 'diced tomato...	[0, 0, 0, 0, 0]	[]
3	alouette potatoes	['spreadable cheese with garlic and herbs', 'n...	[0, 0, 0, 0, 0]	[]
4	amish tomato ketchup for canning	['tomato juice', 'apple cider vinegar', 'sugar...	[0, 0, 0, 0, 0]	[]

How did I classify the food taste?

- So, what should we do then? Can we just drop the unlabeled foods?
- Nope, that's not interesting at all!
- Search engine must have a big database.
- How can we recommend to the user if we only have a few foods?

How did I classify the food taste?

- Since we now have a labeled data, we can train a model to label the unlabeled data.
- What algorithm will we use? Random forest? SVM?
- Remember though, this is some kind of *multiclass classification.* We will classify the food not only to one taste, but might be more than one taste (ex. sweet and spicy, sweet and sour, etc.).
- Besides, we only have food ingredients as the feature. We must convert all of the ingredients into dummy variable, then we can use those algorithms. But, there are so many food ingredients out there! Isn't it going to be sparse?
- So, I don't now if we can use those algorithms directly.

How did I classify the food taste?

- This is when probability theory comes in.
- I don't know whether it's actually a known algorithm, but here's the idea.
- We will identify the taste of each unique ingredients.
- We will not classify them into a specific taste, but rather calculate the probability of the ingredient's each taste.
- For example, I'm 80% sure that "*bittersweet chocolate*" is sweet and 40% sure that "*bittersweet chocolate*" is bitter.
- Still don't get it? Don't worry, I'll show you a more concrete example later.

How did I classify the food taste?

- Warning! Since, we will use probability, there will be some mathematical notations up ahead. So, get ready!
- Let A be the set of all food ingredients, T be the set of food taste, AT_{ij} be the set of food that contains ingredient $a_i \in A$ and the taste is $t_j \in T$, and $f: A \to T$ is a function that maps the ingredient into its taste. Probability of a_i's taste is t_j is

$$P\left(f(a_i) = t_j\right) = \frac{\left|AT_{ij}\right|}{\left|\bigcup_{k=1}^{5} AT_{ik}\right|},$$

where $\left|AT_{ij}\right|$ denotes the total of food that contains a_i and the taste is t_j, and $\left|\bigcup_{k=1}^{5} AT_{ik}\right|$ denotes the total of food that contains a_i.

How did I classify the food taste?

- More confused? Let's take a look at the table below. We will try t calculate the taste's probability of *"bittersweet chocolate"*.

name	ingredients	taste
bittersweet chocolate sorbet	water, granulated sugar, cocoa powder, bittersweet chocolate	sweet, bitter
bittersweet or white chocolate ice cream	milk, granulated sugar, bittersweet chocolate, heavy cream, vanilla extract	sweet, bitter
salty chocolate pecan candy	pecans, bittersweet chocolate, white chocolate, coarse sea salt	salty
a cup of hot mocha michael smith	bittersweet chocolate, butter, eggs, brown sugar, vanilla, strong coffee	sweet, sour
cipriani's chocolate ice cream with bitter orange sauce	milk, bittersweet chocolate, egg yolks, orange zest, fresh orange juice, orange preserves	bitter

How did I classify the food taste?

- Let's assume that *"bittersweet chocolate"* is a_1.
- AT_{11}: foods that contains *"bittersweet chocolate"* and the taste is sweet, i.e. "*bittersweet chocolate sorbet*", "*bittersweet or white chocolate ice cream*", and "*a cup of hot mocha michael smith*".
- AT_{12}: foods that contains *"bittersweet chocolate"* and the taste is salty, i.e. "*salty chocolate pecan candy*".
- AT_{13}: foods that contains *"bittersweet chocolate"* and the taste is sour, i.e. "*a cup of hot mocha michael smith*".
- AT_{14}: foods that contains *"bittersweet chocolate"* and the taste is sweet, i.e. "*bittersweet chocolate sorbet*", "*bittersweet or white chocolate ice cream*", and "*cipriani's chocolate ice cream with bitter orange sauce*".
- AT_{15}: foods that contains *"bittersweet chocolate"* and the taste is spicy, i.e. none.

How did I classify the food taste?

- Probability of *"bittersweet chocolate"* taste is sweet is
$$P(f(a_1) = t_1) = \frac{|AT_{11}|}{\left|\bigcup_{k=1}^{5} AT_{ik}\right|} = \frac{3}{5} = 60\%.$$
- Probability of *"bittersweet chocolate"* taste is salty is
$$P(f(a_1) = t_2) = \frac{|AT_{12}|}{\left|\bigcup_{k=1}^{5} AT_{ik}\right|} = \frac{1}{5} = 20\%.$$
- Probability of *"bittersweet chocolate"* taste is sour is
$$P(f(a_1) = t_1) = \frac{|AT_{13}|}{\left|\bigcup_{k=1}^{5} AT_{ik}\right|} = \frac{1}{5} = 20\%.$$
- Probability of *"bittersweet chocolate"* taste is bitter is
$$P(f(a_1) = t_1) = \frac{|AT_{14}|}{\left|\bigcup_{k=1}^{5} AT_{ik}\right|} = \frac{3}{5} = 60\%.$$
- Probability of *"bittersweet chocolate"* taste is spicy is 0 (it's obvious right? ☺).

How did I classify the food taste?

- Wait! The sum of the probability is not 1! Why!?
- Remember though, this is *multiclass classification.*
- If an ingredient's taste is sweet, it doesn't mean that it's not bitter. It's still probable to be bitter with a high probability.

How did I classify the food taste?

- After we calculate the probability, we can represent the ingredient into its vector representation.
- For example, the vector representation of *"bittersweet chocolate"* is [0.6, 0.2, 0.2, 0.6, 0].
- That means, I'm sure that *"bittersweet chocolate"* might be sweet with 60% confidence, might be salty with 20% confidence, and so on.

```
prob_vectors = []

for ingredient in all_unique_ingredients:
    count_taste = np.zeros(5, dtype = int)
    count_food = 0

    for row in range(len(labeled_data)):
        if ingredient in labeled_data["ingredients"][row]:
            count_taste += np.array(labeled_data["taste"][row])
            count_food += 1

    prob_vector = []
    for i in count_taste:
        prob_vector.append(i/count_food)

    prob_vectors.append(np.array(prob_vector))
```

How did I classify the food taste?

- After we get the vector representation of each ingredients, we take the sum of all ingredients for each unlabeled food and take the average for each taste.
- Then, we take the taste which has probability more than 0.5.
- Confused? Look at the table below.

food	ingredients	sweet	salty	sour	bitter	spicy
bittersweet beachside brownies	bittersweet chocolate	0.6	0.2	0.2	0.6	0
	dark brown sugar	0.8	0.1	0.1	0.5	0
	all-purpose flour	0.3	0	0	0.5	0
Average		0.567	0.1	0.1	0.53	0

How did I classify the food taste?

- Since the average probability of sweet and bitter are more than 0.5, we can say that "*bittersweet beachside brownies*" is sweet and bitter.
- Wow, cool!

food	ingredients	sweet	salty	sour	bitter	spicy
bittersweet beachside brownies	bittersweet chocolate	0.6	0.2	0.2	0.6	0
	dark brown sugar	0.8	0.1	0.1	0.5	0
	all-purpose flour	0.3	0	0	0.5	0
Average		0.567	0.1	0.1	0.53	0

How did I classify the food taste?

- But, why must 0.5?
- Well, it's just a threshold. So, it's up to you on how to choose it. ☺

```
def classify_taste(ingredients):
    ingredients = ingredients.split(", ")
    total_prob = np.zeros(5)
    for ingredient in ingredients:
        if ingredient in all_unique_ingredients:
            total_prob += prob_vectors[all_unique_ingredients.index(ingredient)]

    taste = [0, 0, 0, 0, 0]
    for i in range(5):
        if total_prob[i]/len(ingredients) >= 0.5:
            taste[i] = 1

    return taste
```

How did I classify the food taste?

- Combine the labeled data and the "labeled" unlabeled data, and voila! We get a new large dataset!

- Then, we can use this new dataset to recommend the food based on user's currently preference taste. ☺

```
unlabeled_data["taste"] = unlabeled_data["ingredients"].map(classify_taste)
unlabeled_data["taste_name"] = unlabeled_data["taste"].map(get_taste_name)
unlabeled_data.head()
```

```
complete_data = pd.concat([labeled_data, unlabeled_data]).reset_index(drop = True)
complete_data.head()
```

	name	ingredients	taste	taste_name
0	arriba baked winter squash mexican style	winter squash, mexican seasoning, mixed spice,...	[1, 0, 0, 0, 1]	sweet, spicy
1	a bit different breakfast pizza	prepared pizza crust, sausage patty, eggs, mil...	[1, 0, 0, 0, 1]	sweet, spicy
2	all in the kitchen chili	ground beef, yellow onions, diced tomatoes, to...	[1, 0, 0, 0, 1]	sweet, spicy
3	alouette potatoes	spreadable cheese with garlic and herbs, new p...	[1, 0, 1, 0, 1]	sweet, sour, spicy
4	amish tomato ketchup for canning	tomato juice, apple cider vinegar, sugar, salt...	[1, 0, 1, 0, 1]	sweet, sour, spicy

How did it recommend the food?

How did it recommend the food?

- Well, it's not that hard, so don't worry.
- We only use **dot product**.
- First, we accept the user's query.
- Next, we find what tastes they want.
- For example, user says "I want to eat something sweet and spicy!!!".
- We can represent that sentence into vector, i.e. [1 0 0 0 1].
- Then, we can calculate the dot product of user's vector representation and all of the food taste's vector representation.
- The higher result will be shown to the user.

How did it recommend the food?

- Wait! What if the user says “I want to eat something sweet, but not spicy!!!”? Or what if the user says “I want to eat something sweet, but only a little bit sour.”?
- Well, I’m sorry to say that it’s one of the recommender’s weakness.
- It only detect the taste appear on the user’s query.
- That means, “not spicy” will be interpreted as “spicy”.
- Maybe next time I will improve it. It will be better if you give me the idea. ☺

How did I deploy the food recommender?

How did I deploy the food recommender?

- This is my first time deploying my creation to a web service, so this might look amateur for you. XD
- I chose **Streamlit** to deploy this food recommender, thanks to this tutorial.
- Using **Streamlit** is the easiest and faster way to deploy our data apps.
- Just upload the code to **Github**, create an account on **Streamlit**, then connect the **Github** repository to the **Streamlit**.
- The detail explanation on how to do it can be read on the tutorial and my code on **Github**.

What are this recommender's weaknesses?

What are this recommender’s weaknesses?

- I’m pretty sure that there are still so many weaknesses since this is a very simple food recommender.
- First, we don’t know how accurate our classification method since I didn’t check the accuracy of the model.
- Why? Because we don’t even know whether our first labelling method is correct. We only depend on the food’s name. It will effect our model of course.

What are this recommender’s weaknesses?

- Second, we don’t know the level of the taste.
- We don’t know how sweet or bitter is “*bittersweet chocolate sorbet*”.
- We only know that it is probably sweet and bitter.
- If only we knew the amount of ingredients used to make that food, then maybe we could guess the taste better.
- User could also say something like “I want to eat something sweet, but not too sour and spicy”.

What are this recommender’s weaknesses?

- Third, it doesn’t know the difference between “*not spicy*” and “*spicy*”. Like I said on the previous section, the recommender will recognize “*not spicy*” as “*spicy*”.
- It’s also case sensitive. A little typo on the taste won’t make the recommender work properly.
- For those problem, we need to use NLP. But..maybe next time. ☺

What are this recommender's weaknesses?

- Fourth, since it only uses dot product between the user's taste preference and food's taste vector representation, we can't sort the food.
- We will show everything that matches with user's preference based on our database's order.
- There is no best recommendation or something like that.

What are this recommender's weaknesses?

- Fifth, I must say that the information given by the recommender is not that informative.
- Just giving the name of the food and ingredients is still confusing for the user.
- It will be better of we mention the link to the Food.com.
- But the problem is, I don't know how to do it. XD

What are this recommender’s weaknesses?

- See..there are so many weaknesses this recommender has.
- In the next future, I will try to improve it one step at the time based on your feedback.
- I will also try to make this recommender more interactive by using chatbot!

Well, I think that's all from me. I hope it will inspire many people to create something great. Don't hesitate to ask, comment, or give any suggestion for me.

Thank you. ☺