KitcheNette: Predicting and Ranking Food Ingredient Pairings using Siamese Neural Networks

Donghyeon Park, Keonwoo Kim, Yonggyu Park, Jungwoon Shin and **Jaewoo Kang**[*]
Korea University
{parkdh, akim, yongqyu, jungwoonshin, kangj}@korea.ac.kr

Abstract

As a vast number of ingredients exist in the culinary world, there are countless food ingredient pairings, but only a small number of pairings have been adopted by chefs and studied by food researchers. In this work, we propose KitcheNette which is a model that predicts food ingredient pairing scores and recommends optimal ingredient pairings. KitcheNette employs Siamese neural networks and is trained on our annotated dataset containing 300K scores of pairings generated from numerous ingredients in food recipes. As the results demonstrate, our model not only outperforms other baseline models, but also can recommend complementary food pairings and discover novel ingredient pairings.

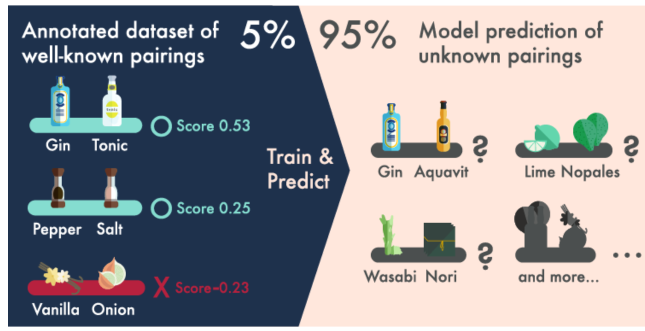


Figure 1: By training on our annotated dataset containing scores of *well-known* food pairings (e.g., gin&tonic water, salt&pepper, vanilla&onion), our model predicts the scores of *unknown* food pairings (e.g., gin&aquavit, wasabi&nori, lime&nopales) that are not annotated because they are less popular or infrequently used.

1 Introduction

Many chefs, gourmets, and food-related researchers have focused on studying food pairing for decades. There are books [Page and Dornenburg, 2008; Dornenburg and Page, 2009] featuring a number of food pairings recommended based on accumulated experiences of professional chefs and food gourmets in the culinary world. Since food pairings are made based on the experiences of experts, food pairing itself is subjective and difficult to quantify. In the academic field, some food-related researchers [Ahn *et al.*, 2011; Ahn and Ahnert, 2013; Garg *et al.*, 2017; Simas *et al.*, 2017] focused on determining the qualities of complementary food pairings based on analysis of sharing flavor compounds. However, FlavorDB built by [Garg *et al.*, 2017] contains only a limited number of flavor compounds and natural ingredients and a considerable amount of time and effort is required to analyze the flavor compounds of food ingredient.

In this work, we introduce KitcheNette which is a model based on Siamese neural networks [Koch *et al.*, 2015]. As shown in Figure 1, KitcheNette first trains on our annotated dataset containing more than 300k scores of *known* pairings, which constitute only 5% of the total possible number of pairings in our dataset. These quantified scores indicate whether each food pair is complementary or not. Then our trained model predicts the scores of *unknown* pairings consisting of food ingredients that have infrequently or never been used in recipes, which constitute 95% of the total number of pairings. The three *unknown* pairings in Figure 1 that our model found are known to be culture specific pairings in Nordic, Japanese, and Mexican cuisine, respectively. To train our model, we constructed our own dataset which contains the scores of 300k food ingredient pairings obtained from 1M human-generated cooking recipes [Salvador *et al.*, 2017; Marin *et al.*, 2018]. Here, the amounts and personal preferences of ingredients and the preparation process in cooking were not considered. Our model employs Siamese neural networks with *wide&deep* architecture designed to learn the relationship of a food pairing. We then conducted experiments to compare it with several baseline models and confirmed that our model KitcheNette outperformed all the other models.

To further evaluate our model's prediction performance, three qualitative analyses were conducted. First, we analyzed some example cases of food pairings to test whether our model successfully predicts the scores of *unknown* pairings. Second, we compared the ranking results of commonly used food ingredient pairings recommended by KitcheNette with those in FlavorDB [Garg *et al.*, 2017]. Our ranking results are more reliable and consistent with human food-pairing knowledge, compared with the results of FlavorDB. Third, we compared the food pairing recommenda-

[*]Corresponding author: Jaewoo Kang (kangj@korea.ac.kr)

tions of our model, which were based on predicted scores, with those of cooking experts [Page and Dornenburg, 2008; Dornenburg and Page, 2009]. We found that most of the recommendations of our model were the same as those of the cooking experts, which demonstrates the accuracy of our model. In addition, our model recommended food pairings with ingredients not commonly featured in recipes. To this extent, our work attempts to broaden the underlying concept of food pairing and introduce a data-driven, deep learning based approach for discovering novel ingredient pairs.

The major contributions are summarized as follows.

- We propose a data-driven task which involves discovering new complementary food pairings that could be used in future recipes.
- We construct a large scale dataset that contains the scores of food ingredient pairings.
- We show that KitcheNette[1] which uses Siamese neural networks and *wide&deep* learning achieves high performance in predicting the scores of food ingredient pairings.
- Through qualitative analyses, we demonstrate KitcheNette's effectiveness in recommending complementary food pairings[2] and discovering new food pairings.

2 Related Work

2.1 Food Related Research

Researches on Discovering Food Pairings. [Ahn *et al.*, 2011] and [Ahn and Ahnert, 2013] introduce a flavor network where the network's edge is built based on the number of flavor compounds shared by culinary ingredients. The flavor network is comprised of 381 ingredients and 1,021 flavor compounds. FlavorDB [Garg *et al.*, 2017] combines existing food repositories to provide a larger database with the user-interactive page. Food-bridging [Simas *et al.*, 2017] improves the flavor network [Ahn *et al.*, 2011] by adding additional bridges between two ingredients through a chain of pairwise affinities despite the chemical compound similarity of the two ingredients being low. However, they cover only a limited number of flavor compounds and natural ingredients and some well-known food pairings (e.g., red wine and beef) have very few flavor compounds in common. Our work employs a data-driven model for predicting the scores of food pairings and finding new food pairings on a larger scale.

Researches on Recommending Recipes. The recipe recommendation [Teng *et al.*, 2012] has been proposed to determine whether a food ingredient is essential in a recipe. recipe recommendation uses two different recipe networks that can accurately predict recipe ratings. Also, finding a surprising and plausible ingredient combination is not new. [Grace *et al.*, 2016; Grace and Maher, 2016] combines cased-based reasoning and deep learning to generate new recipe designs. Our model KitcheNette is similar in that it recommends a new combination of food ingredients, but KitcheNette trains on datasets of pairing scores on a larger scale and can suggest novel food pairings.

2.2 Siamese Neural Networks

Siamese neural networks [Koch *et al.*, 2015] have been employed in various tasks to learn similarities between two different inputs. Also, some variations of Siamese neural networks have been introduced.

[Mueller and Thyagarajan, 2016] proposed the Manhattan LSTM model which takes two sentences as input. The Manhattan LSTM model generates vector representations for each sentence and calculates the similarity between the vector representations using the simple function $exp(-||h1 - h2||)$ where $h1$ and $h2$ are the embedding vectors. [Yuan *et al.*, 2018] proposed a customized contrastive loss function that can be divided into a partial loss function for positive pairs and a partial loss function for negative pairs. The loss function widens the distance between two vector representations of negative pairs while narrowing the distance between two vector representations of positive pairs. While these two works ([Mueller and Thyagarajan, 2016; Yuan *et al.*, 2018]) use Siamese neural networks and take two different inputs of the same type, [Liang *et al.*, 2018]'s proposed Siamese-based model takes one input as the standard by which the other input is evaluated. The input is evaluated based on its similarity to the input used as the standard. Our model is designed to train semantic relationships of a food pair beyond simple distance-based similarity mentioned above.

3 Dataset

3.1 Dataset Description and Preprocessing

In this work, we utilized Recipe1M [Marin *et al.*, 2018], a dataset containing approximately one million recipes and their corresponding images which were collected from multiple popular websites related to cooking. All content in Recipe1M can be divided into two categories: texts and images. The recipe texts of Recipe1M [Marin *et al.*, 2018] consist of the following two parts: the list of ingredients and the instructions of a recipe. The Im2Recipe [Salvador *et al.*, 2017] used a bi-directional LSTM based ingredient name extraction module that performs logistic regression on each word in all the lists of ingredients in Recipe1M to extract the ingredient names only apart. For instance, "2 tbsp of olive oil" is extracted as olive_oil. From the instructions for a recipe, [Marin *et al.*, 2018] trains all the vocabulary including ingredient names with the word2vec [Mikolov *et al.*, 2013] fashion. Among a total of 30,167 learned vocabulary, we obtained 3,567 unique ingredient names as shown in Table 1.

3.2 Food Ingredient Pairing Score Generation

Food Ingredient Pairing Score Dataset. As mentioned in the earlier sections, traditional ingredient pairing methods rely on the extensive experience and knowledge of human experts in the culinary industry. As a result, the amount of data available for training deep-learning models is insufficient. To address this problem, we construct a new large-scale dataset of food pairing scores on which deep-learning models can be trained. Food pairings are given a score from -1 to 1 based on

[1] *https://github.com/dmis-lab/KitcheNette*

[2] A demo version of user-interactive KicheNette is available at *http://kitchenette.korea.ac.kr*.

Recipe1M	# of Recipes	# of Vocab	# of Ingredient Vocab
	1,029,720	30,167	3,567
Ingredient Pairing Dataset	**Total # of Possible Pairs**	**# of *Known* Pairs[I]**	**# of *Unknown* Pairs**
	6,359,961	356,451	6,003,510

Table 1: Statistics of Ingredients and Pairings. *Known Pairs*[I] consist of ingredients whose occurrence counts are greater than 20. In addition, each known pair has a co-occurrence count of at least 5.

Ingredient1 [Count]	Ingredient2 [Count]	Co-occurrence	Pairing score
vanilla [51,756]	baking_soda [58,931]	14,657	0.376
	cocoa [6,520]	2,759	0.360
	powdered_sugar [26,729]	6,558	0.314
	nut [9,090]	2,865	0.312
	chocolate_chips [9,172]	2,821	0.307
	onion [191,691]	12	-0.589
	soy_sauce [40,518]	6	-0.483
	salt_and_pepper [46,534]	14	-0.479
	garlic [46,534]	9	-0.477
	pepper [68,984]	26	-0.462

Table 2: Ingredient Pairing Dataset. The five best&worst ingredients in our dataset for pairing with vanilla. The *The Flavor Bible* [Page and Dornenburg, 2008] recommended pairing chocolate, coffee, cream, ice cream, or sugar with vanilla.

how well the ingredients complement each other. We assumed that the co-occurrence information of ingredients from a large recipe corpus can provide insight into how ingredients are combined in each recipe. In this study, we did not consider the number of ingredients or cooking procedures since our dataset contains statistical co-occurrence information.

Normalized Point-wise Mutual Information. We calculated our food pairing scores based on point-wise mutual information (PMI) as introduced in [Teng *et al.*, 2012]. The PMI score (1) is the probability of two elements co-occurring $p(x, y)$, which is compared to the probability of each element occurring separately $p(x)$, $p(y)$. The custom score is designed to accurately represent good/bad pairs by penalizing highly popular ingredients such as salt or butter with low co-occurrence pairs. On the other hand, a pair consisting of a less popular ingredient is a good, complementary pair (e.g., wasabi&nori). In our work, we used the normalized version of PMI (NPMI (2) [Bouma, 2009]) to better train and fit our regression model. Point-wise mutual information can be normalized between -1 and +1 where -1 (in the limit) is given for never occurring together, 0 for independence, and +1 for complete co-occurrence. Thus, the scores between -1 and 1 intuitively determine if the pair is well suited or not.

$$\text{pmi}(x; y) = \log \frac{p(x, y)}{p(x)p(y)} \tag{1}$$

where:

$$p(x, y) = \frac{\text{\# of recipes where x and y occur together}}{\text{\# of recipes}}$$

$$p(x) = \frac{\text{\# of recipes where x occurs}}{\text{\# of recipes}}, p(y) = \frac{\text{\# of recipes where y occurs}}{\text{\# of recipes}}$$

$$\text{npmi}(x; y) = \frac{\text{pmi}(x; y)}{h(x, y)} \tag{2}$$

where:

$$h(x, y) = -\log p(x, y)$$

Generating Ingredient Pairing Dataset based on NPMI Scores. Ideally, we would calculate all the food ingredient pairing scores of 6,359,961 possible pairing$\binom{3,567}{2}$ generated from 3,567 unique ingredients. However, we found that ingredients that rarely occurred in 1 million recipe texts may act as noisy samples. Also, ingredients that rarely co-occur may lower the performance of the model. Therefore, we removed ingredients whose occurrence count does not exceed 20 and ingredient pairings whose co-occurrence count dose not exceed 5 to build our *known* pairs.

Results. As a result in Table 1, we obtained a total of 356,451 valid *known* pairings. All the other pairings were considered as *unknown* pairings. The final distribution of food ingredient pairing scores follows the approximated normal distribution. We assume that the ingredient pairing scores (0.274 or higher) in the upper 5% ($\mu + 2\sigma$) of the distribution are the complement pairings, whereas scores with negative numbers are examples of bad pairings as shown in Table 2.

4 Model

4.1 Learning Ingredient Representations

We propose a model that predicts the scores of ingredient pairings. Our model architecture consists of two major components, as shown in Figure 2. The first is the 'Ingredient Representation Component' which uses Siamese neural networks [Koch *et al.*, 2015] where two identical multi-layer perceptrons (MLPs) with the same weights each receive a different 300-dimensional word vector representation. Each MLP has two fully connected layers which process the input ingredient vector. Let (X_a, X_b) be a pair of ingredients represented as 300-dimensional word vectors. W_* and b_* are the shared weights and bias of an MLP, respectively, and $f(\cdot)$ denotes the activation function for non-linearity. We use $f(\cdot)$ ($\max(x, 0)$) as rectified linear units (ReLUs). The learned representations (h_a, h_b) of this pair are mathematically expressed as follows:

$$h_a = f(W_2 f(W_1 X_a + b_1) + b_2)$$
$$h_b = f(W_2 f(W_1 X_b + b_1) + b_2)$$

where $W_1 \in \mathbb{R}^{i \times 300}$, $W_2 \in \mathbb{R}^{j \times i}$, $b_1 \in \mathbb{R}^i$, and $b_2 \in \mathbb{R}^j$, and i and j are the number of hidden units.

4.2 Predicting Food Ingredient Pairing Scores

In the 'Pairing Score Prediction Component', we employ *wide&deep* learning [Cheng *et al.*, 2016]. The layer is divided into a *wide layer* and a *deep layer*. In the *deep layer*, two j-dimensional learned representation vectors are concatenated and passed to another MLP that computes a joint representation of two ingredients. This joint representation is denoted as deep vector d and is mathematically expressed as follows:

$$d = f(W_4 f(W_3(h_a, h_b) + b_3) + b_4)$$

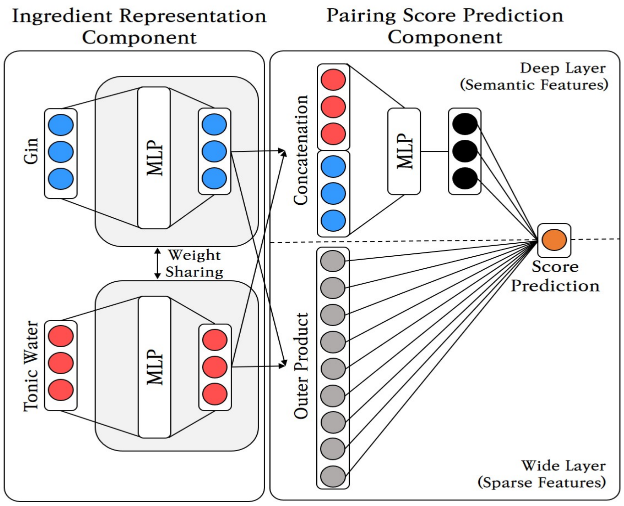


Figure 2: Overview of our KitcheNette architecture

where $W_3 \in \mathbb{R}^{j\times 2j}$, $W_4 \in \mathbb{R}^{j\times j}$, $b_3 \in \mathbb{R}^j$, $b_4 \in \mathbb{R}^j$, and j is the number of hidden units in each layer. In the *wide layer*, the outer product of two j-dimensional learned representation vectors is computed and a $j \times j$ matrix is generated. The matrix is then flattened to a j^2-dimensional *wide vector* w which is mathematically expressed as follows:

$$w = g(h_a \otimes h_b)$$

where g denotes a flattening function that converts a $n \times n$ matrix into a n^2-dimensional single vector.

The *wide vector* w is then directly concatenated to the *deep vector* d. The concatenation of the wide and deep vectors is passed to the last fully connected layer to compute the pair score for the final output. Overall, as (w, d) is the concatenation of the *wide* and *deep* vectors, the final output score Y for the ingredient pair (X_a, X_b) is mathematically calculated as follows:

$$Y = W_5(w, d) + b_5$$

where $W_5 \in \mathbb{R}^{1\times(j^2+j)}$ and $b_5 \in \mathbb{R}^1$, and j is the number of hidden units in each layer.

4.3 Model Training Details

We train our proposed model to minimize the loss function (Mean Squared Error) which can be expressed as follows:

$$L(\Theta) = \frac{1}{N}\sum_{a,b}(y_{ab} - Y_{ab})^2$$

where L is the computed loss function to be minimized during training, Θ are the model parameters to be trained, y_{ab} is the true score value, Y_{ab} is the predicted score value, and N is the total number of input pairs used for training. We use the Adam optimizer for our model.

5 Experiment

5.1 Baseline Models

We first evaluated the baseline models before evaluating our proposed model. We first predicted the pairing scores by simply calculating the cosine similarity between two input ingredient vectors. We then employed the following machine learning models from the Python Scikit-learn [Pedregosa *et al.*, 2011] package as our baseline models: Linear Support Vector Regressor, Random Forest Regressor, Extra Tree Regressor, SGD Regressor, and Gradient Boosting. Additionally, a simple version of Siamese neural networks [Koch *et al.*, 2015] is used one of the baseline models. All these models are fitted with hyperparameters estimated by the built-in grid search.

5.2 Main Results

As illustrated in Table 3, the following five metrics were utilized to evaluate model performance: Root Mean Square Error (RMSE), Mean Square Error (MSE), Mean Absolute Error (MAE), Correlation (CORR), and R squared (R2). Our KitcheNette model clearly outperforms the baseline models in all metrics.

We use Normalized Discounted Cumulative Gain (NDCG@K) to evaluate the ranking performance of our model (Figure 3a) and employ ROC curve to evaluate the sensitivity of our model (Figure 3b). In terms of NDCG@K, our model outperforms all the baseline models in making accurate predictions. The ROC curve is also used to measure the classification performance of the models. As mentioned in Section 3.2, we regarded all pairings with prediction scores of 0.274 or higher as complementary pairings; pairings with lower scores were considered non-complementary. The ROC curve results demonstrate that our KitcheNette model achieves higher performance than all the other models in classifying complementary pairings.

5.3 Ablation Study

We performed ablation tests to evaluate each feature of KitcheNette. As illustrated in Table 4, the *wide&deep* architecture and ingredient embedding of our model help improve its performance. When our model uses the cosine similarity of learned representations from the Siamese networks, it obtains the lowest performance in predicting food pairing scores. The concatenation (*deep layer*) of two representations dramatically improves the performance of our model. This indicates that semantic relations need to be learned for predicting food pairing scores. Furthermore, the *wide&deep* architecture that learns the relation of two ingredients further boosts our model's performance. Also, utilizing the ingredient embedding for input vectors, instead of randomly initialized vectors, improves the model's performance.

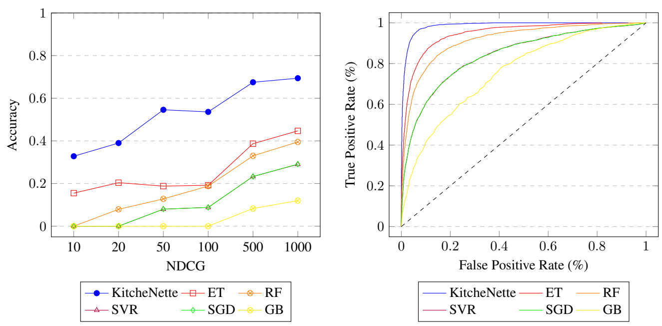
6 Qualitative Analysis

For the qualitative analysis, we considered performing experiments with actual food and obtain human feedback, but realized it would be difficult to evaluate large-scale pairing scores and is beyond the scope of our work. Instead, we performed various case studies. In addition, we provide a demo version[3] of KitcheNette where anyone can retrieve the scores of ingredient pairings.

[3] *http://kitchenette.korea.ac.kr/*

Model	Validation RMSE	Validation MSE	Validation MAE	Validation CORR	Validation R2	Test RMSE	Test MSE	Test MAE	Test CORR	Test R2
Cosine Similarity	-	-	-	-	-	0.1802	0.0325	0.1328	0.3952	-1.6026
Gradient Boosting	0.1073	0.0115	0.0815	0.3339	0.0773	0.1073	0.0115	0.0815	0.3351	0.0776
SGD	0.0993	0.0099	0.0762	0.4585	0.2102	0.0984	0.0097	0.0759	0.4730	0.2236
Linear SVR	0.0993	0.0099	0.0762	0.4588	0.2105	0.0984	0.0097	0.0759	0.4731	0.2238
Random Forest	0.0802	0.0064	0.0612	0.7015	0.4846	0.0799	0.0064	0.0611	0.7042	0.4885
Extra Tree	0.0742	0.0055	0.0566	0.7664	0.5586	0.0738	0.0054	0.0563	0.7689	0.5637
Siamese Network	0.0726	0.0054	0.0540	0.8223	0.5679	0.0729	0.0054	0.0544	0.8235	0.5662
KitcheNette	**0.0421**	**0.0018**	**0.0320**	**0.9249**	**0.8551**	**0.0417**	**0.0018**	**0.0317**	**0.9266**	**0.8583**

Table 3: Prediction results of the models.



(a) NDCG@K (b) ROC Curve

Figure 3: Additional Model Prediction Results.

	RMSE	MSE	MAE	CORR	R2
KitcheNette	**0.0421**	**0.0018**	**0.0320**	**0.9249**	**0.8551**
+Cosine,-Wide&Deep	0.0726	0.0054	0.0540	0.8223	0.5679
-Wide Layer	0.0432	0.0019	0.0326	0.9206	0.8474
-Ingredient Embedding	0.0501	0.0025	0.0383	0.8923	0.7958

Table 4: Ablation tests on the validation set.

6.1 Finding Unknown Pairings

To demonstrate the accuracy of KitcheNette's predictions on infrequently used food ingredient pairings, we performed a comparative analysis of the prediction results of both *known* and *unknown* ingredient pairings. As illustrated in Table 5, we chose three similar carbonated white wines (champagne, sparkling_wine, and prosecco). We then calculated the score of each wine paired with a different ingredient and analyzed all the possible pairings for the three different cases below.

Case 1. We used the champagne&orange_twist as a given pairing for comparison since it is a well-known ingredient pairing with a high annotated score. The orange_twist&sparkling_wine and orange_twist&prosecco pairings do not have annotated scores since they are uncommon pairings. Additionally, we chose two other ingredients (orange_wedge and lime_twist) that are similar to orange_twist, but are not frequently used with any of the three wines. As a result, we have one *known* pairing and eight *unknown* pairings. The prediction results of all the nine pairings were consistently high (0.33-0.45).

Case 2. Based on the annotated scores, we paired the wines with ingredients that are different but complement at the same time with them, to create the following three unique *known* pairings: champagne&elderflower_liqueur, sparkling_wine&cream_de_cassis, and prosecco&lemon_sorbet). The prediction results of the remaining six *unknown* pairings were also consistently high (0.29-0.42) compared to the three given *known* pairings.

		champagne	sparkling_wine	prosecco
Case 1	orange_twist	0.33†	0.39*	0.42*
	orange_wedge	0.37*	0.43*	0.45*
	lime_twist	0.34*	0.38*	0.40*
Case 2	elderflower_liqueur	0.34†	0.39*	0.41*
	creme_de_cassis	0.29*	0.33†	0.34*
	lemon_sorbet	0.32*	0.39*	0.42†
Case 3	onion	-0.20†	-0.14*	-0.17*

Table 5: Examples of *known* and *unknown* pairings. † and * refer to the predicted scores of *known* and *unknown* pairings, respectively

Case 3. Finally, we chose onion as it made the worst *known* pairing with champagne and paired it with the remaining two wines. The predictions results of the two *unknown* pairings (sparkling_wine&onion and prosecco&onion) were consistently as low as the scores of the *known* pairing (champagne&onion).

Results. In sum, our prediction results show that KitcheNette is capable of making predictions on certain pairings based on analogical reasoning which states that if A is similar to B and A forms a good pairing with C, then it is more likely that B also forms a good pairing with C. We believe that using this reasoning enhanced the performance of KitcheNette on *unknown* food pairings without annotated scores.

6.2 Comparison of Food Pairing Ranking Results

We performed a comparative analysis between the ingredients ranked by KitcheNette and the ranked ingredients in FlavorDB[4]. We selected four widely used food ingredients (tomato, onion, pepper and cinnamon) and retrieved the top 10 ingredient pairings to each selected ingredient as shown in Table 6. Based on our observations, KitcheNette generally recommended food ingredients that are frequently and actually used in everyday cooking and dining (e.g., tomato&lettuce, onion&ground_beef, pepper&oregano, cinnamon&clove,apple). On the other hand, while FlavorDB

[4] *https://cosylab.iiitd.edu.in/flavordb/search*

	Tomato		Onion		Pepper		Cinnamon	
Rank	*KitcheNette*	*FlavorDB*	*KitcheNette*	*FlavorDB*	*KitcheNette*	*FlavorDB*	*KitcheNette*	*FlavorDB*
1	lettuce	tea	bay_leaf	cocoa	oregano	ginger	allspice	pepper
2	avocado	potato	celery	garlic	ground_beef	laurel	clove	ginger
3	cucumber	mango	ground_beef	peanut	potato	rosemary	raisin	laurel
4	bean_dip	guava	potato	potato	thyme	basil	baking_soda	basil
5	eggplant	apple	carrot	tomato	elbow_macaroni	orange	apple	rosemary
6	turmeric_powder	grape	tomato_paste	chive	basil	spearmint	nutmeg	nutmeg
7	garam_masala	soybean	beef_broth	soybean	celery	oregano	applesauce	oregano
8	red_chili_powder	strawberry	beef_stock	green_beans	onion	nutmeg	brown_sugar	cassia
9	tostados	cocoa	green_pepper	tea	hamburger	celery	pumpkin_puree	tea
10	taco_shells	mushroom	stewing_beef	leek	marjoram	dill	canned_pumpkin	celery

Table 6: The ranking results of KitchenNette and FlavorDB[Garg *et al.*, 2017].

	Red Wine		White Wine		Gin		Sake	
Rank	*KitcheNette*	*Recommendations*	*KitcheNette*	*Recommendations*	*KitcheNette*	*Recommendations*	*KitcheNette*	*Recommendations*
1	beef_stock	beef	mussel	butter	angostura_bitters	apple brandy	mirin	Japanese cusine
2	beef_cheeks	cheeze	shad*	chicken	sweet_vermouth	apricot brandy	katakuriko	cucumber
3	lamb_shank	game	cockle	crab	benedictine	basil	dashi_stock	fish
4	beef_broth	lamb	shrimp_shells	cream	orange_bitters	blackberries	konnyaku	gin
5	pan_juices*	meat, red	fish_fumet	fish	elderflower_liqueur	celery	burdock_root	lemon juice
6	chicken_backs	peper, black	lobster_base	lobster	orange_twist	champange	miso	salads
7	saltpeter*	steak	arborio_rice	salmon	lemon_twist	cilantro	soy_sauce	sashimi and sushi
8	tomato_paste	starawberries	skate*	scallops	simple_syrup	cola	gochujang	shellfish
9	oxtail		oyster_liquor*	shellfish	dry_vermouth	cranberry juice	mitsuba	sugar
10	dry_red_wine		cuttlefish*	veal	orange_wedge*	ginger	dashi	vodka
11	veal_stock		shrimp		aquavit*	herbs	pork_belly	
12	ajinomoto*		fish_stock		pisco*	lemon juice	umeboshi	
13	pike*		mirlitons*		rye_whiskey*	lime juice	kombu	
14	lamb_stock		crayfish*		honey_syrup	mint	okonomiyaki_sauce*	
15	beef_bones*		escargot		mezcal*	orange juice	bonito_flakes	
16	verjuice*		scampi*		absinthe	oysters	yuzu	
17	cherry_cola*		clam_juice		curacao	tonic	white_sesame_seeds	
18	beef_stew_seasoning*		fish_bones		campari		wood_ear_mushrooms	
19	brisket*		lobster_shells*		lemon_twists		daikon_radish	
20	ti_leaves*		scallop		wheat_starch*		kamaboko	

Table 7: Food&drink Pairings. The ranked pairings of KitchenNette and food&drink recommendations from *"The Flavor Bible"*[Page and Dornenburg, 2008] and *"WHAT to DRINK with WHAT you EAT"*[Dornenburg and Page, 2009]. The recommendations are listed in alphabetical order. * refers to the predicted scores of *unknown* pairings.

recommended food ingredients that share a large number of chemical compounds with the selected ingredients, some of the recommendations did not pair well with the selected ingredients (e.g., tomato&tea, onion&cocoa, pepper&orange) for cooking and dining.

6.3 Discovering New Drink-Food Pairings

We also found that KitcheNette can discover new food-drink pairings, which we believe is one of the main aims of food pairing. As illustrated in Table 7, we compared our model's food&drink recommendations with those from *"The Flavor Bible"* [Page and Dornenburg, 2008] and *"WHAT to DRINK with WHAT you EAT"* [Dornenburg and Page, 2009]. We found that KitcheNette not only provides recommendations that are consistent with the recommendations of culinary experts from the books, but also recommends far more pairings than the two books.

For red wine and white wine, our model recommended a variety of meat (e.g., beef, lamb) and specific seafood ingredients (e.g., mussel, lobster, shrimp) respectively. Our model recommended authentic Japanese food ingredients to pair with sake, which shows that our data-driven learning model is also able to recommend food ingredients less common in non-Asian cuisines.

7 Conclusion & Future Work

In this work, we introduced KitcheNette which predicts food ingredient pairing scores and ranks them based on the predicted scores. Our model which has Siamese neural networks is trained on dataset containing more than 300k food ingredient pairing scores. We demonstrate that our model discovers new and unknown pairings and achieves better ranking results than the existing food pairing ranking models.

For future work, we plan to use a graph-based neural network architecture to train on one-to-many ingredient pairings, instead of on one-to-one pairings, which were used by our model's Siamese networks. Also, we plan to add the chemical information of food ingredients to the ingredient embeddings and use more detailed information on food ingredients from food encyclopedias. Last, we would like to use more novel and authentic recipes to help our model to recommend more versatile food ingredient pairings.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the government of Korea (MSIP) (NRF-2017R1A2A1A17069645, NRF-2017M3C4A7065887)

References

[Ahn and Ahnert, 2013] Yong-Yeol Ahn and Sebastian Ahnert. The flavor network. *Leonardo*, 46(3):272–273, 2013.

[Ahn *et al.*, 2011] Yong-Yeol Ahn, Sebastian E Ahnert, James P Bagrow, and Albert-László Barabási. Flavor network and the principles of food pairing. *Scientific reports*, 1:196, 2011.

[Bouma, 2009] Gerlof Bouma. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, pages 31–40, 2009.

[Cheng *et al.*, 2016] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 7–10. ACM, 2016.

[Dornenburg and Page, 2009] Andrew Dornenburg and Karen Page. *What to Drink with What You Eat: The Definitive Guide to Pairing Food with Wine, Beer, Spirits, Coffee, Tea-Even Water-Based on Expert Advice from America's Best Sommeliers*. Little, Brown, 2009.

[Garg *et al.*, 2017] Neelansh Garg, Apuroop Sethupathy, Rudraksh Tuwani, Shubham Dokania, Arvind Iyer, Ayushi Gupta, Shubhra Agrawal, Navjot Singh, Shubham Shukla, Kriti Kathuria, et al. Flavordb: a database of flavor molecules. *Nucleic acids research*, 46(D1):D1210–D1216, 2017.

[Grace and Maher, 2016] Kazjon Grace and Mary Lou Maher. Surprise-triggered reformulation of design goals. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[Grace *et al.*, 2016] Kazjon Grace, Mary Lou Maher, David C Wilson, and Nadia A Najjar. Combining cbr and deep learning to generate surprising recipe designs. In *International Conference on Case-Based Reasoning*, pages 154–169. Springer, 2016.

[Koch *et al.*, 2015] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, volume 2, 2015.

[Liang *et al.*, 2018] Guoxi Liang, Byung-Won On, Dongwon Jeong, Hyun-Chul Kim, and Gyu Choi. Automated essay scoring: A siamese bidirectional lstm neural network architecture. *Symmetry*, 10(12):682, 2018.

[Marin *et al.*, 2018] Javier Marin, Aritro Biswas, Ferda Ofli, Nicholas Hynes, Amaia Salvador, Yusuf Aytar, Ingmar Weber, and Antonio Torralba. Recipe1m: A dataset for learning cross-modal embeddings for cooking recipes and food images. *arXiv preprint arXiv:1810.06553*, 2018.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[Mueller and Thyagarajan, 2016] Jonas Mueller and Aditya Thyagarajan. Siamese recurrent architectures for learning sentence similarity. In *AAAI*, volume 16, pages 2786–2792, 2016.

[Page and Dornenburg, 2008] Karen Page and Andrew Dornenburg. *The flavor bible: The essential guide to culinary creativity, based on the wisdom of America's most imaginative chefs*. Little, Brown, 2008.

[Pedregosa *et al.*, 2011] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[Salvador *et al.*, 2017] Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, and Antonio Torralba. Learning cross-modal embeddings for cooking recipes and food images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[Simas *et al.*, 2017] Tiago Simas, Michal Ficek, Albert Diaz-Guilera, Pere Obrador, and Pablo R Rodriguez. Food-bridging: a new network construction to unveil the principles of cooking. *Frontiers in ICT*, 4:14, 2017.

[Teng *et al.*, 2012] Chun-Yuen Teng, Yu-Ru Lin, and Lada A Adamic. Recipe recommendation using ingredient networks. In *Proceedings of the 4th Annual ACM Web Science Conference*, pages 298–307. ACM, 2012.

[Yuan *et al.*, 2018] Huiru Yuan, Guannan Liu, Hong Li, and Lihong Wang. Matching recommendations based on siamese network and metric learning. In *2018 15th International Conference on Service Systems and Service Management (ICSSSM)*, pages 1–6. IEEE, 2018.