Duplicated Cooking Recipe Determination Using Multimodal Information

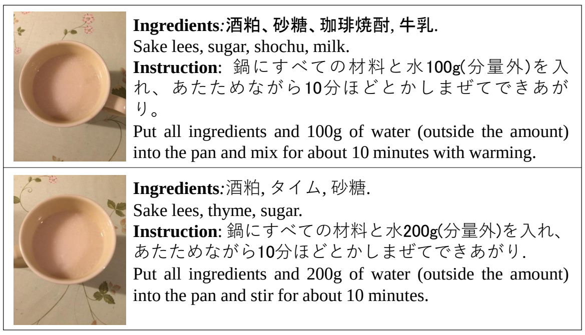
Tung The Nguyen[1] Yuki Nakayama[2]
[1]Nara Institute of Science and Technology
[2]Rakuten Institute of Technology, Rakuten Inc.
[1]nguyen.tung.np5@is.naist.jp, [2]yuki.b.nakayama@rakuten.com

Abstract There are many online platforms that allow people to upload and share cooking recipes with each other. However, within those platforms, many recipes contain the same contents, which can cause many problems and waste a lot of resources. In this work, we tackle the problem of duplicated cooking recipe determination by consider it as a binary classification task, in which we need to classify a pair of recipes into either duplicate or non-duplicate. We proposed a method based on multilayer perceptron (MLP). Our model takes input features from multiple modalities, which are extracted from the input recipes. We conducted experiments on the Rakuten Recipe database[1]and observed that the proposed method outperformed the baseline and achieved very high performance for the duplicate recipe determination task.

1 Introduction

Duplicated cooking recipe is becoming a big problem for online recipe sharing websites. It causes wasteful usage of storage since we still need to hold the duplicated recipes in the database. Besides, troubles can arise when a user is looking for a certain recipe and two duplicated recipes appear in the search results. Let us assume that those two recipes have different titles (which is a common case) and both of them do not match the user's needs. In this situation, the user needs to check both of the recipes, which costs them time and burden unnecessarily. Finally, since the recipe sharing platforms usually reward the users who post cooking recipes with benefits, there are many users who exploit this reward system by uploading duplicated recipes to gain rewards unfairly. If we can detect the duplication, we can tell which user has this kind of behavior and prevent them from doing it. In general, a cooking recipe database contains millions of recipes, which makes manual duplication checking infeasible. Therefore, it is essential for us to find a way to perform automatically detect duplication of cooking recipes efficiently and accurately.

The task of detecting duplicated cooking recipes can be divided into two types: internal and external recipe duplication detection. With the first type (internal), we need to identify duplicated recipes within a given database. For the external type, assume that we have a set of cooking recipes; given a new recipe, we need to determine whether it is duplicated with any recipe in the provided database or not. Our work focuses on the duplicated recipe determination task, which is defined as follow: *given two recipes, determine if they are duplicated or not.* Obviously, this task is necessary to perform both internal and external recipe duplication detection.



Ingredients*:*酒粕、砂糖、珈琲焼酎, 牛乳.
Sake lees, sugar, shochu, milk.
Instruction: 鍋にすべての材料と水100g(分量外)を入れ、あたためながら10分ほどとかしまぜてできあがり。
Put all ingredients and 100g of water (outside the amount) into the pan and mix for about 10 minutes with warming.

Ingredients*:*酒粕, タイム, 砂糖.
Sake lees, thyme, sugar.
Instruction: 鍋にすべての材料と水200g(分量外)を入れ、あたためながら10分ほどとかしまぜてできあがり.
Put all ingredients and 200g of water (outside the amount) into the pan and stir for about 10 minutes.

Figure 1: *Example of two duplicated cooking recipes.*

We can view the duplicated recipe determination task as performing a classification of the two input recipes into either duplicate or non-duplicate. Multilayer perceptron is a popular and powerful model for classification [5]; thus, we decided to use it for building our classification model. Figure 1 shows an example of duplicated cooking recipes. These two are similar in terms of the images, the instruction, and the ingredients. Therefore, we decided to use them as input features for our classification model. In addition, we observed that duplicated recipes are usually uploaded by the same user; therefore, our model also uses this information for duplication determination. In summary, our proposed method is

[1] https://recipe.rakuten.co.jp/

based on multimodal approach, in contrast to previous works [1, 7] that use single-modality features. Details of the proposed method will be explained in Section 3.

We conducted experiments of duplicated recipe determination using Rakuten Recipe database. Our proposed method outperformed the baseline significantly in multiple metrics and reached more than 95% of detection accuracy. The contribution of our work is two-fold:

1. This work is one of the first to utilize deep neural network classification model for deciding whether two cooking recipes are duplicated or or not.

2. We utilize features from multiple modalities for the duplicated recipe determination task.

2 Related works

The task of duplicated cooking recipe determination is related to document plagiarism detection. Heres proposed a method [2] for detecting document plagiarism using classification by deep neural network. In this work, the model only uses linguistic features, which are the embedding vectors of the documents as input. On the other hand, a cooking recipe contains information from multiple modalities such as image, cooking instruction, list of ingredients, and other kinds of metadata information such as user or food tags. Our method takes advantages of all these information and uses them as input features.

Oguni et al. [7] also tackles the problem of external duplicated recipe detection. However, this work does not provide a clear solution for the duplicated recipe determination task. In addition, this work only uses features from a single modality (either image or cooking instruction text) for the duplication detection, even though this work extracts features from multiple modalities. As a result, the performance reported in [7] was not really high. Another method proposed by Hanai et al.[1] for finding similar recipes using clustering based on ingredients. The reported results of this work were also low performance and not enough to be used in practical situations.

3 Method

Our proposed method can be divided into two parts: feature extraction and classification model, which are described in the following sections.

3.1 Feature Extraction

We extract features from the two input recipes in terms of linguistic (cooking instruction), visual (food images), and metadata information (ingredient lists and users). The extracted features reflect the similarity between the input recipes for each modalities, namely, linguistic, visual, and metadata.

For linguistic features, first, we encode the cooking instruction text into embedding vector by using Sparse Composite Document Vector – SCDV [3]. After that, a similarity score in terms of instruction between two input cooking recipes is calculated by measuring the Euclidean distance of the two embedding vectors. To encode the food image, we use the pre-trained Inception-v3 embedding model, which is widely used for object detection [6]. Similarly, we use Euclidean distance of the resultant embedding vectors as the similarity score of the food images from input recipes. For measuring the similarity between ingredients used in the two recipes, first, we calculate the Jaccard distance between two lists of ingredients. Next, we subtract this value from 1 and the result is used as a similarity score in terms of ingredients between the two given recipes. Finally, with the user similarity score, we use a function that takes value of 0 if the user ID from two recipes are the same and 1 otherwise.

We can see that each feature used in our method is a single value that represents similarity of the two input recipes in terms of multiple modalities. We did not use the raw embedding vectors as in [2] because they are high-dimensional, thus increase computational cost significantly compared to our method.

3.2 Classification Model using Multilayer Perceptron

Multilayer perceptron is a simple but powerful classification model that is being used in various tasks. Because of this reason, we use MLP for building our classification model. An illustration of the proposed method is shown in Figure 2.

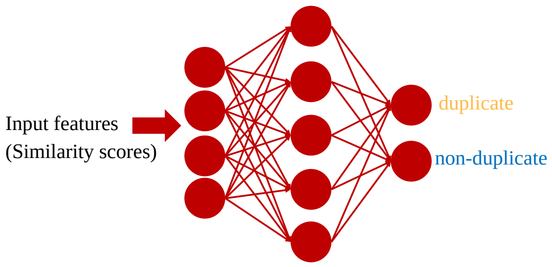


Figure 2: *The classification model of the proposed method.*

Our classification model takes input as a vector containing features described above and classifies it into two classes: duplicate and non-duplicate. Our proposed method can be considered as multimodal approach, because input features contain information from multiple modalities: linguistic, visual, and metadata. We expect that the proposed method can achieve higher performance compared to single-modality based approach.

4 Experiments

The dataset we used in these experiments was taken from the Rakuten Recipe sharing platform. From this database, we picked up 1,857 pairs of recipes that closely resemble each other in terms of similarity score. We manually annotated the pairs with a label of “duplicate” or “non-duplicate”. After that, we split the dataset into training, development, and test set. The statistics of our dataset are shown in Table 1.

Table 1: Dataset statistics.

	Training	Development	Test
# of pairs in total	1547	100	200
# of duplicate	1255	50	100
# of non-duplicate	292	50	100

Our classification model contains only one hidden layer, as shown in Figure 2. Empirically, we observed that increasing the number of hidden layer in the network to be two or more did not improve performance. We trained the model using Adam optimizer. The development set was used for tuning the learning rate and number of units in the hidden layer. Particularly, the learning rate is $1e-3$ and the size of hidden layer is 32.

In the first experiment, we compared our proposed method with a baseline based on Oguni et al.[7]. As mentioned above, this work did not provide a clear solution for the recipe duplication determination task. In this work, the authors ranked recipe pairs based on their similarity scores either in terms of cooking instruction or the food image. After that, the top k recipe pairs with highest scores are regarded as duplicates.

Table 2: Results of duplicate recipe determination. Rec = recall, Pre = precision.

		Positive label			Negative label		
Baseline	k	Rec	Pre	F1	Rec	Prec	F1
	60	0.47	0.78	0.59	0.62	0.87	0.73
	80	0.54	0.68	0.60	0.60	0.74	0.67
	100	0.64	0.64	0.64	0.64	0.64	0.64
	120	0.67	0.56	0.61	0.59	0.47	0.52
Proposed		**0.99**	**0.92**	**0.95**	**0.91**	**0.99**	**0.95**

In accordance this method, we built a baseline that uses cooking instruction similarity score for ranking, as shown in Table 2. Positive label indicates duplicate and negative label indicates non-duplicate pairs. At every value of k, the proposed method significantly outperforms the baseline in all performance metrics ($p < 0.05$). When we changed to a baseline that uses similarity score of food image, the results remained similar.

In the next experiment, we conducted ablation tests to investigate the importance of the features described in section 3.1. The results of this experiment are summarized in Table 3.

Table 3: Feature importance in duplicate recipe determination. Rec = recall, Pre = precision.

Features	Positive label			Negative label		
	Rec	Pre	F1	Rec	Pre	F1
Instruction + Ingredient	0.85	0.66	0.74	0.56	0.79	0.66
Instruction + Ingredient + User	0.94	0.90	0.92	0.90	0.94	0.92
Instruction + Ingredient + Image	0.97	0.91	0.94	0.90	0.97	0.93
All features	**0.99**	**0.92**	**0.95**	**0.91**	**0.99**	**0.95**

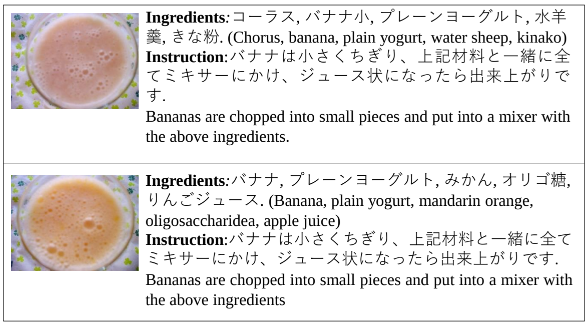
Table 3 shows that using information from the user and the food image significantly improved classification performance from just using cooking instruction and ingredient features. Thus, other than the currently used information from cooking instruction and ingredients, user and food image are also effective features for determining duplicate recipes. Obviously, the highest performance was achieved when using features from all modalities.

Table 4: Confusion matrix of classification.

Ground truth Prediction	Duplicate	Non-duplicate
Duplicate	99	9
Non-duplicate	1	91

Table 4 shows the confusion matrix of the classification results of our proposed model. Our model can predict correctly almost all of the duplicated recipe pairs (99 out of 100). On the other hand, the majority of mis-classification errors are false positive, where the model mistakenly identified a non-duplicate pair as duplicate.

We further investigated errors of our model by checking the mis-classified recipe pairs. Because most of the errors are false positive, we focused on analyzing this type of error. Figure 3 shows an example of such classification mistake. We saw that even though this pair is labeled as non-duplicate and contains differences in ingredients, the two recipes bear striking resemblances, having exactly the same in-



Ingredients: コーラス, バナナ小, プレーンヨーグルト, 水羊羹, きな粉. (Chorus, banana, plain yogurt, water sheep, kinako)
Instruction:バナナは小さくちぎり、上記材料と一緒に全てミキサーにかけ、ジュース状になったら出来上がりです.
Bananas are chopped into small pieces and put into a mixer with the above ingredients.

Ingredients:バナナ, プレーンヨーグルト, みかん, オリゴ糖, りんごジュース. (Banana, plain yogurt, mandarin orange, oligosaccharidea, apple juice)
Instruction:バナナは小さくちぎり、上記材料と一緒に全てミキサーにかけ、ジュース状になったら出来上がりです.
Bananas are chopped into small pieces and put into a mixer with the above ingredients

Figure 3: *Example of classification error.*

struction text and slightly different images; thus, our model classified this pair as duplicate. We observed that this phenomenon also appears in other false positive classification errors. One plausible solution is to fine-tune the visual embedding model Inception-v3 using food images from Rakuten Recipe. The fine-tuning process will allow us to generate embedding vectors that are better at reflecting the differences in food images.

5 Conclusion

In this work, we proposed a new method for determining duplicate cooking recipes. Our proposed method is based on multilayer perceptron model and uses multimodal information for classification. Results from experiments showed that our proposed method significantly outperformed the baseline.

In the future, we would like to investigate the performance of the model when using more powerful multimodal fusion method such as hierarchical tensor fusion network [4]. We also plan to apply the proposed method for comparison of other type of documents that contains multimodal information such as online articles or academic papers.

Acknowledgment

We would like to thank Koji Murakami from Rakuten Institute of Technology - New York for the insightful discussions and the help of checking this manuscript.

References

[1] Shunsuke Hanai, Hidetsugu Nanba, and Akiyo Nadamoto. Clustering for closely similar recipes to extract spam recipes in user-generated recipe sites. In *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services*, p. 31. ACM, 2015.

[2] Daniël Heres. Source code plagiarism detection using machine learning. Master's thesis, 2017.

[3] Dheeraj Mekala, Vivek Gupta, Bhargavi Paranjape, and Harish Karnick. Scdv: Sparse composite document vectors using soft clustering over distributional representations. *arXiv preprint arXiv:1612.06778*, 2016.

[4] Tung The Nguyen, Koichiro Yoshino, Sakriani Sakti, and Satoshi Nakamura. Hierarchical tensor fusion network for deception handling negotiation dialog model. In *The third conversational AI workshop - NeurIPS 2019*, 2019.

[5] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[6] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

[7] 小邦将輝, 平手勇宇, 関洋平. 調理手順テキストと料理画像の特徴量の最近傍探索に基づく重複レシピの検出手法 (ヒューマンコミュニケーション基礎). 電子情報通信学会技術研究報告= IEICE technical report: 信学技報, Vol. 118, No. 278, pp. 19–24, 2018.