Cross-modal Recipe Retrieval: How to Cook This Dish?

Jingjing Chen, Lei Pang, and Chong-Wah Ngo

Department of Computer Science, City University of Hong Kong
{jingjchen9-c, leipang3-c}@my.cityu.edu.hk
cscwngo@cityu.edu.hk

Abstract. In social media users like to share food pictures. One intelligent feature, potentially attractive to amateur chefs, is the recommendation of recipe along with food. Having this feature, unfortunately, is still technically challenging. First, the current technology in food recognition can only scale up to few hundreds of categories, which are yet to be practical for recognizing ten of thousands of food categories. Second, even one food category can have variants of recipes that differ in ingredient composition. Finding the best-match recipe requires knowledge of ingredients, which is a fine-grained recognition problem. In this paper, we consider the problem from the viewpoint of cross-modality analysis. Given a large number of image and recipe pairs acquired from the Internet, a joint space is learnt to locally capture the ingredient correspondence from images and recipes. As learning happens at the region level for image and ingredient level for recipe, the model has ability to generalize recognition to unseen food categories. Furthermore, the embedded multi-modal ingredient feature sheds light on the retrieval of best-match recipes. On an in-house dataset, our model can double the retrieval performance of DeViSE, a popular cross-modality model but not considering region information during learning.

Keywords: Recipe retrieval, cross-modal retrieval; multi-modality embedding

1 Introduction

Food recognition is generally regarded as a hard problem, due to diverse appearances of food as a result of non-rigid deformation and composition of ingredients. Recently, the problem has started to capture more attention [1] [2] [3] [4] partly due to the success of deep learning technologies. The accuracy of food recognition can be as high as 80% on the benchmark datasets such as Food101 [2], FoodCam-256 [5] and VIREO Food-172 [6]. The success gives light to the development of techniques for auto dietary food tracking [1] [7] [8] and nutrition estimation [9], which has long been recognized as a challenge not only in multimedia [8] [10] but also health and nutritional science [11].

Nevertheless, the existing efforts are mostly devoted to recognizing a pre-defined set of food categories, ranging from 100 to 256 categories [2] [3] [5] [6]. Extending to large-scale recognition, for example tens of thousands food categories, remains an area yet to be researched. In this paper, we pose food recognition as a problem of recipe retrieval. Specifically, given a food picture, of whether the category has been seen in the training model, the aim is to retrieve a recipe for the food. The advantages of having recipe, rather than the name of food category, as output are numerous. Sharing food

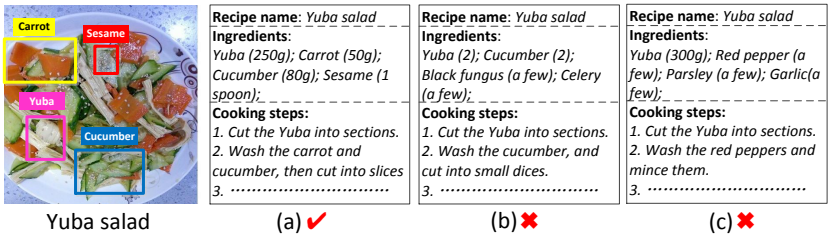

Fig. 1. Although recipe (a), (b) and (c) are all about "Yuba salad", only recipe (a) uses the exactly same ingredients as the dish picture. Retrieving best-match recipe requires fine-grained analysis of ingredient composition.

pictures in social media has been a trend. The ability to recommend recipes along will benefit users who want to cook a particular dish, and the feature is yet to be available. In addition, recipe provides rich information, such as cooking methods, ingredients and their quantities, which can facilitate the estimation of food balance and nutrition facts. The challenge of recipe retrieval, nevertheless, comes from the fact that there could be many recipes named under the same categories, each of which differs in the composition of ingredients. Figure 1 shows an example, where recommending the right recipe for "Yuba Salad" indeed requires also fine-grained recognition of ingredient composition.

This paper explores the recent advances in cross-modality learning for addressing the aforementioned problems. Specifically, given food pictures and their associated recipes, our aim is to learn a model that captures their correspondence by learning a joint embedding space for visual-and-text translation. We exploit and revise a deep model, stacked attention network (SAN) [12], originally proposed for visual question-answering for our purpose. The model learns the correspondence through assigning heavier weights to the attended regions relevance to ingredients extracted from recipes. For the task of recipe retrieval, fortunately the learning does not require much effort in labeling training examples. There are already millions of food-recipe pairs, uploaded by professional and amateur chefs, on various cooking websites, which can be freely leveraged for training. We demonstrate that using these online resources, a fairly decent model can be trained for recipe retrieval with minimal labeling effort. As input to SAN includes ingredients, the model has higher generalization ability in recognizing food categories unseen during training, as long as all or most ingredients are known. Furthermore, as ingredient composition is considered in SAN, the chance of retrieving the best-match recipes is also enhanced. To this end, the contribution of this paper lies in addressing of food recognition as a recipe retrieval problem. Under this umbrella, the problem is turned into cross-modality feature learning, which can integrally model three inter-related problems: scalable food recognition, fine-grained ingredient recognition and best-match recipe retrieval.

2 Related work

Analysis of recipes has been studied from different perspectives, including retrieval [6] [13] [14], classification [15] [16] and recommendation [17]. Most of the approaches

employ text-based analysis based upon information extracted from recipes. Examples include extraction of ingredients as features for cuisine classification [15] and taste estimation [16]. More sophisticated approaches model recipes as cooking graphs [13] [18] such that graph-based matching can be employed for similarity ranking of recipes. The graph, either manually or semi-automatically constructed from a recipe, represents the workflow for cooking and cutting procedures of ingredients. In [13], multi-modality information was explored, by late fusion of cooking graphs and low-level features extracted from food pictures, for example-based recipe retrieval. Few works have also studied cross-modality retrieval [6] [14] [17]. In [17], recognition of raw ingredients was studied for cooking recipe recommendation. Compared to prepared food where ingredients are mixed or even occlude each other, raw ingredients are easier to recognize. In [14], classifier-based approach was adopted for visual-to-text retrieval. Specifically, the category of food picture is first recognized, followed by retrieval of recipes under a category. As classifiers were trained from UPMC Food-101 dataset [2], retrieval is only limited to 101 food categories. The issues in scalability and finding best-match recipes are not addressed. The recent work in [6] explored ingredient recognition for recipe retrieval. Using ingredient network as external knowledge, the approach is able to retrieve recipes even for unseen food categories. Different from [6], this paper aims to learn a joint space that can inherently capture the visual-text commonality for retrieval.

Cross-modality analysis has been actively researched for multimedia retrieval [19] [20] [21]. Frequently employed algorithms include canonical correlation analysis (CCA) [22] and partial least squares (PLS) [23], which find a pair of linear transformation to maximize the correlation between data from two modalities. CCA, in particular, has been extended to three-view CCA [24], semantic correlation matching (SCM) [19], deep CCA [25] and end-to-end deep CCA [26] for cross-modality analysis. Among variants of model, deep visual semantic embedding (DeViSE) [20] is generally used and usually exhibits satisfactory performance. These models, nevertheless, consider image-level features, such as *fc7* extracted from deep convolutional network (DCNN), and usually ignore regional features critical for fine-grained recognition. One of the exceptions is deep fragment embedding (DFE) proposed in [21], which aligns image objects and sentence fragments while learning the visual-text joint feature. However, the model is not applicable here for requiring of R-CNN [27] for object region detection. In food domain, there is yet to have any algorithm for robust segmentation of ingredients, which can be fed into DFE for learning.

3 Stacked Attention Network (SAN)

Figure 2 illustrates the SAN model, with visual and text features respectively extracted from image and recipe as input. The model learns a joint space that boosts the similarity between images and their corresponding recipes. Different from [12], where the output layer is for classification, we modify SAN so as to maximize the similarity for image-recipe pairs. As SAN considers spatial information, attention map can be visualized by back projection of embedded feature into image.

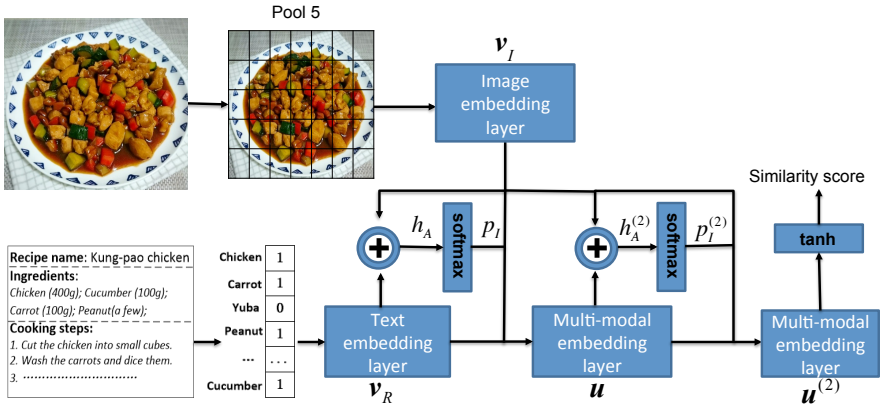

Fig. 2. SAN model inspired from [12] for joint visual-text space learning and attention localization.

3.1 Image Embedding Feature

The input visual feature is the last pooling layer of DCNN – *Pool5* – that retains the spatial information of the original image. The dimension of *Pool5* feature is $512 \times 14 \times 14$, corresponding to 14×14 or 196 spatial grids of an image. Each grid is represented as a vector of 512 dimensions. Denote f_I as the *Pool5* feature and is composed of regions f_i, $i \in [0, 195]$. Each region f_i is transformed to a new vector or embedding feature as following:

$$\boldsymbol{v}_I = tanh(W_I f_I + b_I) \tag{1}$$

where $\boldsymbol{v}_I \in \mathbb{R}^{d \times m}$ is the transformed feature matrix, with d as the dimension of new vector and $m = 196$ is the number of grids or regions. The embedding feature of f_i is indexed by i-th column of $\boldsymbol{v}_I$, denoted as $\boldsymbol{v}_i$. The transformation is performed region-wise, $W_I \in \mathbb{R}^{d \times 512}$ is the transformation matrix and $b_I \in \mathbb{R}^d$ is the bias term.

3.2 Recipe Embedding Feature

A recipe is represented as a binary vector of ingredients, denoted as $\boldsymbol{r} \in \mathbb{R}^t$. The dimension of vector is t corresponding to the size of ingredient vocabulary. Each entry in $\boldsymbol{r}$ indicates the presence (1) or absence (0) of a particular ingredient in a recipe. As Pool5 feature, the vector is embedded into a new space as following

$$\boldsymbol{v}_R = tanh(W_R \boldsymbol{r} + b_r) \tag{2}$$

where $W_R \in \mathbb{R}^{d \times t}$ is the embedding matrix and $b_r \in \mathbb{R}^d$ is the bias vector. Note that, for joint learning, the embedding features of recipe ($\boldsymbol{v}_R \in \mathbb{R}^d$) and Pool5 region (i-th column of $\boldsymbol{v}_I$) have the same dimension.

3.3 Joint embedding feature

The attention layer is to learn the joint feature by trying to locate the visual food regions that correspond to ingredients. There are two transformation matrices, $W_{I,A} \in \mathbb{R}^{k \times d}$

for image I and $W_{R,A} \in \mathbb{R}^{k \times d}$ for recipe R, mimicking the attention localization, formulated as following:

$$h_A = tanh(W_{I,A}\boldsymbol{v}_I \oplus (W_{R,A}\boldsymbol{v}_R + b_A)) \tag{3}$$

$$p_I = softmax(W_P h_A + b_P) \tag{4}$$

where $h_A \in \mathbb{R}^{k \times m}$, $p_I \in \mathbb{R}^m$, $W_P \in \mathbb{R}^{1 \times k}$. Note that p_I aims to capture the attention, or more precisely relevance, of image regions to a recipe. The significance of a region f_i is indicated by the value in the corresponding element $p_i \in p_I$.

The joint visual-text feature is basically generated by adding the embedding features v_I and v_R. To incorporate attention value, regions v_i are linearly weighted and summed (equation-5) before the addition operation with v_R (equation-6), as following:

$$\tilde{\boldsymbol{v}_I} = \sum_{i=1}^{m} p_i v_i \tag{5}$$

$$\boldsymbol{u} = \tilde{\boldsymbol{v}_I} + \boldsymbol{v}_R \tag{6}$$

where $\tilde{\boldsymbol{v}_I} \in \mathbb{R}^d$, and $\boldsymbol{u} \in \mathbb{R}^d$ represents the joint embedding feature.

As suggested in [12], progressive learning by stacking multiple attention layers can boost the performance, but will heavily increase the training cost. We consider two-layer SAN, by feeding the output of first attention layer, $\boldsymbol{u}^{(1)}$, into the second layer to generate new joint embedding feature $\boldsymbol{u}^{(2)}$ as following

$$h_A^{(2)} = tanh(W_{I,A}^{(2)}\boldsymbol{v}_I \oplus (W_{R,A}^{(2)}\boldsymbol{u} + b_A^{(2)})) \tag{7}$$

$$p_I^{(2)} = softmax(W_P^{(2)} h_A^{(2)} + b_P^{(2)}) \tag{8}$$

$$\tilde{\boldsymbol{v}_I^{(2)}} = \sum_i p_i^{(2)} v_i \tag{9}$$

$$\boldsymbol{u}^{(2)} = \tilde{\boldsymbol{v}_I^{(2)}} + \boldsymbol{u} \tag{10}$$

As $p_I^{(2)}$ indicates the region relevancy, the attention map can be visualized by back projecting the attention value p_i to its corresponding region f_i, followed by upsampling to the original image size with bicubic interpolation.

3.4 Objective Function

To this end, the similarity between food image and recipe is generated as following:

$$S < \boldsymbol{v}_I, \boldsymbol{v}_R >= tanh(W_{u,s}\boldsymbol{u}^{(2)} + b_s) \tag{11}$$

where $W_{u,s} \in \mathbb{R}^d$ and $b_s \in \mathbb{R}$ is bias. $S < \boldsymbol{v}_I, \boldsymbol{v}_R >$ outputs a score indicating the association between the embedding features of image and recipe. The learning is based on the following rank-based loss function with a large margin form as the objective function:

$$\mathcal{L}(W, D_{trn}) = \sum_{(\boldsymbol{v}_I, \boldsymbol{v}_R^+, \boldsymbol{v}_R^-) \in D_{trn}} max(0, \triangle + S < \boldsymbol{v}_I, \boldsymbol{v}_R^- > - S < \boldsymbol{v}_I, \boldsymbol{v}_R^+ >) \tag{12}$$

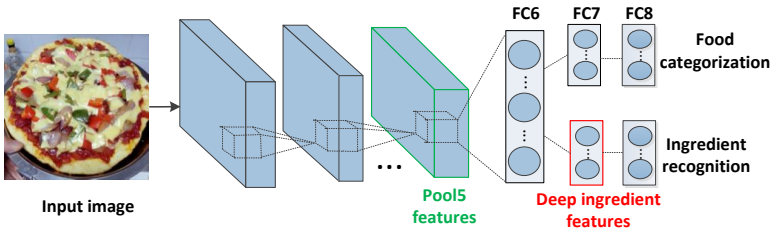

Fig. 3. Multi-task VGG model in [6] offering *pool5* and deep ingredient features for cross-modal joint space learning.

The training set, D_{trn}, consists of triples in the form of $(\boldsymbol{v}_I, \boldsymbol{v}_R^+, \boldsymbol{v}_R^-)$, where $\boldsymbol{v}_R^+$ ($\boldsymbol{v}_R^-$) is true (false) recipe for food v_I. The matrix W represents the network parameters, and $\triangle \in (0, 1)$ controls the margin in training and is cross-validated.

4 Experiments

4.1 Settings and Evaluation

Here we detail the parameter setting of SAN. The dimension of embedding feature is set to $d = 500$ for both *Pool5* regional and recipe feature, while the dimension for h_A is $k = 1,024$ for equations 3 and 7. Through cross-validation, the hyper parameter $\triangle$ for the loss function is set as 0.2. SAN is trained using stochastic gradient descent with momentum set as 0.9 and the initial learning rate as 1. The size of mini-batch is 50 and the training stops after 10 epochs. To prevent overfitting, dropout [28] is used. The *pool5* feature can be extracted from any DCNN models. We employed the multi-task VGG released by [6], which reported the best performances on two large food datasets, VIREO Food-172 [6] and UEC Food-100 [3]. The model, as shown in Figure 3, has two pathways, one for classifying 172 food categories while another for labeling 353 ingredients. For a fair comparison, all the compared approaches in the experiment are using multi-task VGG features, either *pool5* or deep ingredient feature (*fc7*), as shown in Figure 3.

As the task is to find the best possible recipe given a food picture, the following two measures are employed for performance evaluation:

- Mean reciprocal rank (MRR): MRR measures the reciprocal of rank position where the ground truth recipe is returned, averaged over all the queries. This measure assesses the ability of the system to return the correct recipe at the top of the ranking. The value of MRR is within the range of [0, 1]. A higher score indicates a better performance.
- Recall at Top-K (R@K): R@K computes the fraction of times that a correct recipe is found within the top-K retrieved candidates. R@K provides an intuitive sense of how quickly the best recipe can be located by investigating a subset of the retrieved items. As MRR, a higher score also indicates a better performance.

4.2 Dataset

The dataset is composed of 61,139 image-recipe pairs crawled from the "Go Cooking"[1] websites. Each pair consists of a recipe and a picture of resolution 448×448. The dataset covers different kinds of food, like Chinese dishes, snacks, dessert, cookies and Chinese-style western food. Each recipe includes the list of ingredients and cooking procedure. As the recipes were uploaded by amateurs, the naming of ingredients is not always consistent. For example, "carrot" is sometimes called as "carotte". We manually rectified the inconsistency and compiled a list of 5,990 ingredients, both visible and non-visible (e.g., "honey"), from these recipes. The list, represented as a binary vector indicating the presence or absence of particular ingredients in a recipe, serves as input to the SAN model. Note that in some cases the cooking and cutting methods are directly embedded into the name of ingredient, for example, "tofu" and "tofu piece", "egg" and "steamed egg".

The dataset is split into three sets: 54,139 pairs for training, 2,000 pairs for cross-validation, and 5,000 pairs for testing. Furthermore, we selected 1,000 images from the testing set as queries to search against the 5,000 recipes. The queries are sampled in such a way that there are around 45% of them (446 queries) belonging to food categories unknown to SAN and multi-task VGG models. In addition, around 85% of the queries have more than one relevant recipe. We recruited a homemaker, who has cooking experience, to manually pick the relevant recipes for each of the 1,000 queries. The homemaker was instructed to label relevant recipes based on title similarity in recipes, titles that are named differently because of geography regions or sharing almost the same cooking procedure with similar key ingredients. For example, the dish "sauteed tofu in hot and spicy sauce" is sometimes called as "mapo tofu" in the restaurant menu. In the extreme case, some queries have more than 60 relevant recipes. On average each query has 9 number of relevant recipes. Note that the testing queries are designed in these ways so as to verify the two major claims in this paper, i.e., the degree in which the learnt model can generalize to unseen food categories (Section 4.4) and the capability in finding the best-matched recipe (Section 4.5).

4.3 Performance Comparison

We compared SAN to both shallow and deep models for cross-modal retrieval as following. The inputs to these models are the deep ingredient feature (*fc7*) of multi-task VGG model and the ingredient vector of 5,990 dimensions. The *Pool5* feature is not used due to high dimensionality ($14 \times 14 \times 512$). As reported in [29], simply concatenating the features from 14×14 grids performs worse than *fc7* in visual recognition.

- Canonical Correlation Analysis (CCA)[22]: CCA is a classic way of learning latent subspace between two views or features by maximizing the correlation between them. Two linear mapping functions are learnt for projections of features into subspace.

[1] https://www.xiachufang.com

- Partial Least Squares (PLS)[23]: Similar to CCA, PLS learns two linear mapping functions between two views. Instead of using cosine similarity as in CCA, PLS uses dot product as the function for measuring correlation.
- DeViSE [20]: DeViSE is a deep model with two pathways which respectively learn the embedded features of recipe-image pairs to maximize their similarities. Note that, instead of directly using word2vec as in [20], the embedded feature of ingredients is learnt from the training set of our dataset. This is simply because word2vec is learnt from documents such as news corpus [30] and lacks specificity in capturing information peculiar to ingredients. Different from SAN, DeViSE is not designed for attention region localization.
- DeViSE++: We purposely included a variant of DeViSE, which takes the hand-cropped regions of food as input to the deep model. The cropping highlights the target food region and basically removes the background or irrelevant part of food pictures. The aim of using DeViSE++ is to gate the potential improvement over DeViSE when only food region is considered, and more importantly, to justify the merit of SAN in identifying appropriate attention region in comparison to hand-cropped region.

Table 1. MRR and R@K for recipe retrieval. The best performance is highlighted in bold font.

Method	MRR	R@1	R@5	R@10	R@20	R@40	R@60	R@80	R@100
CCA	0.055	0.023	0.079	0.123	0.182	0.262	0.329	0.371	0.413
PLS	0.032	0.009	0.039	0.073	0.129	0.219	0.284	0.338	0.398
DeViSE	0.049	0.016	0.060	0.108	0.182	0.300	0.391	0.456	0.524
DeViSE++	0.05	0.016	0.059	0.105	0.174	0.307	0.404	0.471	0.531
SAN	**0.115**	**0.048**	**0.161**	**0.249**	**0.364**	**0.508**	**0.60**1	**0.671**	**0.730**

Table 1 lists the results of different approaches. Deep models basically outperform shallow models in terms of recall at the depth of 20 and beyond. In contrast to PLS, which does not perform score normalization, CCA manages to outperform DeViSE in terms of MRR and R@K for $K < 20$. Among all these approaches, the proposed model SAN consistently exhibits the best performance across all the measures. Compared to DeViSE, SAN achieves a relative improvement of 130% in MRR and doubles its performance at R@20, which is fairly impressive.

Despite the encouraging performance by SAN, the value of R@1 is only around 0.05. Figure 4 shows some successful and near-miss examples. The first two pictures show query images where all visible ingredients are clearly seen. SAN manages to retrieve the ground-truth recipe at top-1 rank in such cases. In the third example, SAN ranks "grilled salmon" higher than "fried salmon" as the current model does not consider cooking attributes. In addition, SAN overlooks the beef and peanuts which are mixed and partially occluded by salmon, while confused by the ingredients of similar appearance, i.e., caviar and red pepper, bean sprout and basil. The last query image shows an example of how non-visible ingredients, flour in this example, affect the ranking. The flour is used to make the dish into round shape, and this knowledge does not seem to be learnt by SAN.

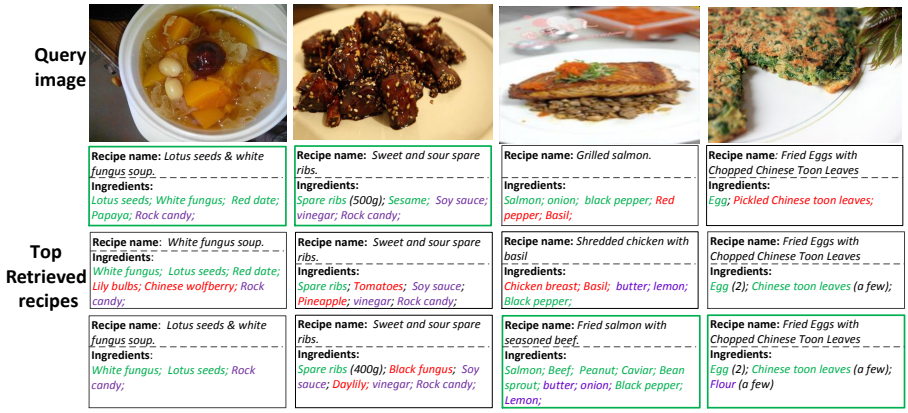

Fig. 4. Examples of top-3 retrieved recipes (ranked from top to bottom). Ground-truth recipe is marked in green. The ingredients in different colours have different meanings: green – true positive, purple – true positive but non-visible in dish, red – false positive.

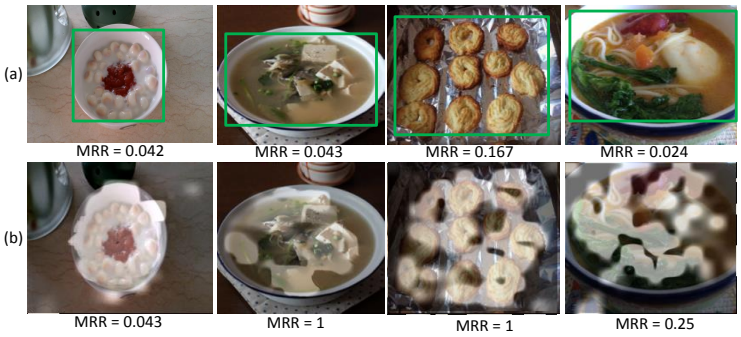

Fig. 5. (a) Examples contrasting the manually cropped region (green bounding box), (b) the learnt attention region (masked in white) by SAN.

Another result worth noticing is that there is no performance difference between DeViSE and DeViSE++. While DeViSE is not designed for attention localization, the model seems to have the ability to exclude irrelevant background regions from recognition. To provide further insights, Figure 5 shows some examples visualizing the attention regions highlighted by SAN and in contrast to hand-crafted regions. In the first example, the region attended by SAN is about the same as the region manually cropped. In this case, DeViSE+ and SAN use to have similar performance. The next two examples highlight the superiority of SAN in excluding soup and foil as attention regions, which cannot be not easily done by simple region cropping. SAN significantly outperforms DeViSE in such examples. Finally, the last example shows a typical case that SAN only highlights part of dishes as attention. While there is no direct explanation of why certain food regions are ignored by SAN for joint space learning, it seems that SAN has the ability to exclude regions that are vague and hard to be recognized even by human.

4.4 Finding the best matches recipes

Recalled that around 85% of query images have more than one relevant recipe. This section examines the ability of SAN in identifying the best (or ground-truth) recipe from the testing set composed of 5,000 recipes. To provide insights, we select the queries that retrieval at least one relevant recipe (excluding ground-truth recipe) within the top-5 position for analysis. We divide the selected queries into 7 groups based on the number of relevant recipes. Table 2 lists the performance. As can be seen from the table, the difficulty of finding best-match is proportional to the number of relevant recipes. Compared of DeViSE, SAN generally shows better performance for R@1. As the number of recipes increases, they tie in performance. Nevertheless, while looking deeper into the list, SAN consistently outperforms DeViSE in terms of R@5 and R@10. Two main reasons that ground truth recipe are not ranked higher are due to occluded ingredients and use of different non-visible ingredients. Two such examples include the last two pictures in Figure 4.

Table 2. Performance comparison between SAN and DeViSE in retrieving best-match recipes.

		R@1		R@5		R@10	
Recipe #	Query #	SAN	DeViSE	SAN	DeViSE	SAN	DeViSE
2-3	33	0.21	0.15	0.67	0.48	0.82	0.76
4-7	66	0.18	0.17	0.56	0.53	0.70	0.67
8-11	54	0.17	0.15	0.54	0.30	0.60	0.50
11-15	38	0.13	0.08	0.47	0.39	0.63	0.55
16-30	48	0.06	0.06	0.46	0.39	0.62	0.52
31-61	25	0.08	0.08	0.28	0.26	0.44	0.44

4.5 Generalization to unknown categories

Table 3 further shows the performance of SAN to unseen categories. As expected, the performance is not as good as that for the food categories known to SAN and multi-task VGG. When the ingredients of unknown food categories are previously seen and can be correctly identified, SAN performs satisfactorily. In contrast, when some ingredients, especially key ingredients, are unknown, the model will likely fail to retrieval relevant recipes.

5 Conclusion

We have presented a deep model for learning the commonality between image and text at the fine-grained ingredient level. The power of model comes from the ability to infer attended regions relevant to ingredients extracted from recipes. This peculiarity enables retrieval of best-match recipes even for unseen food category. The experimental results basically verify our claims that the model can deal with unknown food categories to the extent that at least key ingredients are seen during the training. In addition, SAN exhibits consistently better performance than DeViSE, showing the advantage of fine-grained ingredient analysis at the regional level for best-match recipe retrieval.

Table 3. Generalization of SAN to unseen food categories.

	Query #	MRR	R@1	R@5	R@10	R@20	R@40	R@60	R@80
Known Category	554	0.125	0.054	0.175	0.263	0.394	0.535	0.623	0.698
Unknown Category	446	0.103	0.04	0.143	0.231	0.327	0.475	0.572	0.637

The current model can be extended to explicitly model cooking attributes, which could address some limitations identified in the experiments. In addition, as the attention layers couple both visual and text features, the embedding features cannot be offline indexed and have to be generated on-the-fly when the query image is given. This poses limitation on retrieval speed for online application, which is an issue needs to be further researched.

6 Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 61272290), and the National Hi-Tech Research and Development Program (863 Program) of China under Grant 2014AA015102.

References

1. Austin Meyers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin P Murphy. Im2calories: towards an automated mobile vision food diary. In *ICCV*, pages 1233–1241, 2015.
2. Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *ECCV*, pages 446–461. 2014.
3. Y. Matsuda, H. Hoashi, and K. Yanai. Recognition of multiple-food images by detecting candidate regions. In *ICME*, 2012.
4. Oscar Beijbom, Neel Joshi, Dan Morris, Scott Saponas, and Siddharth Khullar. Menu-match: Restaurant-specific food logging from images. In *WACV*, pages 844–851, 2015.
5. Yoshiyuki Kawano and Keiji Yanai. Foodcam-256: A large-scale real-time mobile food recognitionsystem employing high-dimensional features and compression of classifier weights. In *ACM MM*, pages 761–762, 2014.
6. Jingjing Chen and Chong-Wah Ngo. Deep-based ingredient recognition for cooking recipe retrieval. In *ACM MM*, 2016.
7. Keigo Kitamura, Toshihiko Yamasaki, and Kiyoharu Aizawa. Food log by analyzing food images. In *ACM MM*, pages 999–1000, 2008.
8. Kiyoharu Aizawa and Makoto Ogawa. Foodlog: Multimedia tool for healthcare applications. *IEEE MultiMedia*, 22(2):4–8, 2015.
9. Weiyu Zhang, Qian Yu, Behjat Siddiquie, Ajay Divakaran, and Harpreet Sawhney. Snap-n-eat: Food recognition and nutrition estimation on a smartphone. *Journal of diabetes science and technology*, 9(3):525–533, 2015.
10. Ruihan Xu, Luis Herranz, Shuqiang Jiang, Shuang Wang, Xinhang Song, and Ramesh Jain. Geolocalized modeling for dish recognition. *TMM*, 17(8):1187–1199, 2015.
11. Yasmine Probst, Duc Thanh Nguyen, Megan Rollo, and Wanqing Li. mhealth diet and nutrition guidance. *mHealth*, 2015.

12. Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. *arXiv preprint arXiv:1511.02274*, 2015.
13. Haoran Xie, Lijuan Yu, and Qing Li. A hybrid semantic item model for recipe search by example. In *Multimedia (ISM), 2010 IEEE International Symposium on*, pages 254–259, 2010.
14. Xin Wang, Devinder Kumar, Nicolas Thome, Matthieu Cord, and Frederic Precioso. Recipe recognition with large multimodal food dataset. In *ICMEW*, pages 1–6, 2015.
15. Han Su, Ting-Wei Lin, Cheng-Te Li, Man-Kwan Shan, and Janet Chang. Automatic recipe cuisine classification by ingredients. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pages 565–570, 2014.
16. Hiroki Matsunaga, Keisuke Doman, Takatsugu Hirayama, Ichiro Ide, Daisuke Deguchi, and Hiroshi Murase. Tastes and textures estimation of foods based on the analysis of its ingredients list and image. In *New Trends in Image Analysis and Processing–ICIAP 2015 Workshops*, pages 326–333, 2015.
17. Takuma Maruyama, Yoshiyuki Kawano, and Keiji Yanai. Real-time mobile recipe recommendation system using food ingredient recognition. In *Proceedings of the ACM international workshop on Interactive multimedia on mobile and portable devices*, pages 27–34, 2012.
18. Yoko Yamakata, Shinji Imahori, Hirokuni Maeta, and Shinsuke Mori. A method for extracting major workflow composed of ingredients, tools and actions from cooking procedural text. In *8th Workshop on Multimediafor Cooking and Eating Activities*, 2016.
19. Nikhil Rasiwasia, Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Gert RG Lanckriet, Roger Levy, and Nuno Vasconcelos. A new approach to cross-modal multimedia retrieval. In *ACM MM*, pages 251–260, 2010.
20. Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, pages 2121–2129, 2013.
21. Andrej Karpathy, Armand Joulin, and Fei Fei Li. Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*, pages 1889–1897, 2014.
22. David R Hardoon, Sandor Szedmak, and John Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12):2639–2664, 2004.
23. Roman Rosipal and Nicole Krämer. Overview and recent advances in partial least squares. In *Subspace, latent structure and feature selection*, pages 34–51. Springer, 2006.
24. Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *IJCV*, 106(2):210–233, 2014.
25. Galen Andrew, Raman Arora, Jeff A Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *ICML*, pages 1247–1255, 2013.
26. Fei Yan and Krystian Mikolajczyk. Deep correlation for matching images and text. In *CVPR*, pages 3441–3450, 2015.
27. Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.
28. Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
29. Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, pages 647–655, 2014.
30. T Mikolov and J Dean. Distributed representations of words and phrases and their compositionality. *NIPS*, 2013.