R^{2}GAN: Cross-modal Recipe Retrieval with Generative Adversarial Network

Bin Zhu[1], Chong-Wah Ngo[1], Jingjing Chen[2], and Yanbin Hao[1]

[1]City University of Hong Kong
binzhu4-c@my.cityu.edu.hk, {cscwngo, yanbihao}@cityu.edu.hk
[2]National University of Singapore
chenjingjing@nus.edu.sg

Abstract

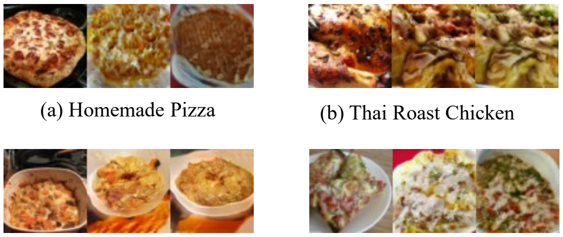
Representing procedure text such as recipe for cross-modal retrieval is inherently a difficult problem, not mentioning to generate image from recipe for visualization. This paper studies a new version of GAN, named Recipe Retrieval Generative Adversarial Network (R^{2}GAN), to explore the feasibility of generating image from procedure text for retrieval problem. The motivation of using GAN is t-wofold: learning compatible cross-modal features in an adversarial way, and explanation of search results by showing the images generated from recipes. The novelty of R^{2}GAN comes from architecture design, specifically a GAN with one generator and dual discriminators is used, which makes the generation of image from recipe a feasible idea. Furthermore, empowered by the generated images, a two-level ranking loss in both embedding and image spaces are considered. These add-ons not only result in excellent retrieval performance, but also generate close-to-realistic food images useful for explaining ranking of recipes. On recipe1M dataset, R^{2}GAN demonstrates high scalability to data size, outperforms all the existing approaches, and generates images intuitive for human to interpret the search results.

1. Introduction

Food is fundamental to health and social participation. Due to abundant food images and recipes available online, food computing for healthcare has recently captured numerous research attentions [34, 22]. Managing to retrieve the recipe of food intake, for example, can assist the estimation of nutrition consumption and hence benefit food logging [22, 5]. The past efforts on food computing range from food categorization [19, 20, 21], food attribution recognition [3, 4, 23], zero-shot recipe retrieval [3] to food perception [36, 27] and recommendation [9, 8, 39].

This paper studies food-to-recipe and recipe-to-food retrieval, which is a typical problem of cross-modal retrieval [38] but peculiar to the domain of food computing. Specifically, recipe is a text article describing preparation of food material and procedure of cooking. A typical recipe consists of three sections: title, ingredients, and cooking instructions, which may or may not align with the visual appearance of a cooked dish. For instance, some ingredients (e.g., sugar, salt) are not visible in dish. Furthermore, cooking instruction more often implies the cause-and-effect of cooking rather than visually depicting the dish appearance. The nature of problem conflicts with the assumption made by the existing cross-modal retrieval, which trains model using text narration that explicitly refers to visual content [31, 32, 18]. Modeling lengthy procedure text such as recipe can thus be a new challenge for cross-modal retrieval.

In the literature, the problem of food-to-recipe retrieval is addressed by either classification [3, 4] or cross-modal learning [35, 2]. Classification-based approaches annotate rich food attributes (e.g., ingredients, cooking and cutting methods) in food images and then match these attributes against words extracted from recipes for retrieval [4]. A major drawback is the significant efforts required in labeling of food attributes, which are not only cost expensive and labour intensive. Cross-modal learning smartly alleviates this requirement, by training latent space that can accommodate both image and text modalities for similarity measurement. The labeling efforts are significantly reduced by requiring only recipe-image pairs, which are easy to collect, than to painstakingly annotate visual food attributes [4]. To model text description in recipe, neural networks of different complexities have been investigated in [35, 5] to learn embeddings for different sections of a recipe. Although efficient, cross-modal learning is inherently an unexplainable model compared to classification-based approaches, which are able to list out the matched attributes as evidences to re-



(a) Homemade Pizza

(b) Thai Roast Chicken

(c) Tater Tot Casserole

(d) Mushroom & Salami Grill

Figure 1. Examples of thumbnails generated by R^2GAN. From left to right are original image, and two thumbnails generated from image and recipe embeddings respectively.

count the retrieval result.

This paper addresses the limitation of cross-modal learning for recipe retrieval. Specifically, a novel deep architecture is designed to interpret cross-modal matching, by synthesizing thumbnail images from recipes to assist the browsing of search results. The machine-generated thumbnails represent how a system perceives the effect of cooking and visually provides cue to explain the ranking of a recipe. Figure 1 shows the examples of thumbnails generated from recipes. As observed, these thumbnails (right) are not only similar to the examples (middle) generated from image embedding, but also the original images (left).

The proposed architecture is built upon cross-modal embedding [35] and generative adversarial network (GAN) [10]. Note that GAN has not yet been studied for this problem. Due to the use of GAN for Recipe Retrieval, we name the proposed model as R^2GAN. As recipes are rich of procedure descriptions, conventional GAN with one generator and one discriminator turns out to be ineffective. As a consequence, R^2GAN is designed to have two discriminators, with one to guess between real and fake images as in common practice, and the other to predict the source of embedding, i.e., whether a fake image is generated from image or recipe embedding. Leveraging on the images generated from different modalities, a novel two-level rank loss function is designed to consider losses in both embedding and image spaces. The overall design of R^2GAN is to encompass a rich set of functions to quantify cross-modal embedding, image reconstruction, food semantics and adversarial losses. With these, R^2GAN is capable of learning *compatible* embeddings for image-to-recipe similarity measure, and performing recipe-to-image generation to *explain* the rationale of similarity.

The main contribution of this paper is exploration of GAN for cross-modal recipe retrieval. Despite the wide use of GAN in various problem domains [30, 40, 37, 41], GAN surprisingly remains not attempted for recipe retrieval. Using GAN, this paper novelly utilizes image generation to visualize what is preserved in a recipe embedding for the explanation of search results. To the best of our knowledge, the proposed R^2GAN with one generator and two discriminators is a relatively new idea. Although the design of dual discriminators has been recently investigated by D2GAN [26], the purpose is to address the issue of mode collapse by combining Kullback-Leibler (KL) and reverse KL divergences into a unified objective function in optimization, which is completely different from this paper. R^2GAN aims for cross-modal learning and its dual discriminators, in contrast to D2GAN, are designed to be functionally different aiming to learn compatible embeddings and explainable thumbnails jointly.

2. Related Works

The core problem of cross-modal retrieval is to measure the similarity between two modalities. Learning common feature subspace is currently the main stream of research [38]. The approaches range from canonical correlation analysis (CCA) [31, 29], which learns subspace to maximize correlation between modalities, to the most recent stacked cross attention model [17], which discovers the full latent alignment to capture fine-grained relationship across modalities. This section focuses on works relevant to food computing.

2.1. Recipe and Food Retrieval

Stacked attention model was first studied in [6] for image-to-recipe retrieval. By representing ingredients extracted from recipe as a binary vector, the model attends to image regions with salient ingredients for learning common latent space. This work, nevertheless, explores only ingredients and cannot disambiguate recipes with the same ingredients list but different cooking procedures. Joint neural embedding (JNE) addresses this problem by proposing bi-directional LSTM to embed the sparse list of ingredients and a hierarchical LSTM to encode the lengthy and complex descriptions of cooking procedure [35]. In addition, regularization with semantic loss, specifically to enforce the learnt embedding to predict food category, is found to be crucial in feature learning. The recent work in [5] improves JNE by introducing title encoder and multi-level attention modeling of cooking instructions from word-level to sentence-level. The new model is capable of assigning lower weights to visually insignificant words, such as "classic" and "home-made", resulting in better retrieval accuracy. Built upon JNE [35], AdaMine recently proposed in [2] surpasses the performances of [35, 5] with large margin, by proposing a double-triplet learning scheme and an adaptive strategy for informative triplet mining. The adaptive strategy is effective in alleviating the problem of gradient diminishing, and hence is also adopted by R^2GAN.

Classification-based approaches are also studied for this problem. In [3], ingredients are multi-labeled on food images to match recipes for retrieval. As only a limited number of 353 ingredients is trained for recognition, the idea of zero-shot recipe retrieval is introduced to retrieve recipes with ingredients unknown to a training model. The problem is addressed by constructing a large graph with both known and unknown ingredients as nodes. The graph models the co-occurrence relationship among ingredients, and conditional random field (CRF) is employed to propagate the prediction scores from known to unknown ingredients for recipe retrieval. This approach, nevertheless, is effective when only a small number of unknown ingredients is considered in the graph. The approach is later extended in [4] by predicting cooking and cutting attributes in addition to ingredients when matching with keywords extracted from recipes. Comparing to cross-modal retrieval, classification-based model is explainable as attributes are explicitly evaluated to quantify the final similarity score. However, training classification models to sufficiently cover a wide variety of food attributes for retrieval is practically intractable.

2.2. Cross-modal GAN

GAN has been applied for generating food images [13], but not in the context of cross-modal learning. In [13], conditioned on food category and ingredients respectively, CGAN [24] is employed to synthesize novel dish images. However, recipes information, including cooking style and process, has not yet been explored.

GAN has captured a lot of research attentions [1, 25, 41, 40, 15]. Although GAN has not been studied for recipe retrieval, cross-modal GAN is not a new idea. Examples include ACMR [37], GXN [11] and CM-GANS [28], with the common goal of learning embedding features for cross-modal retrieval. Different from most GANs, ACMR [37] does not have generator to reconstruct image. Instead, features are generated from images or text captions for the discriminator to guess the source of modality, which is similar to the second discriminator of R^2GAN. GXN [11] has two pairs of generator-discriminator, where a generator synthesizes examples of different modalities for discriminator to guess between real and fake samples. CM-GANS [28], different from ACMR and GXN, considers a whole paragraph of text instead of a short sentence in learning. CM-GANS also has two pairs of generator-discriminator for image-to-image and text-to-text generation. Similar to ACMR, cross modal learning is enabled by having a discriminator to predict the modality of an embedded feature. Having two pairs of generator-discriminator is not considered in R^2GAN because generating procedure description from image is practically implausible. Instead, the design of pairing one generator with dual discriminators is adopted. Different from ACMR and CM-GANS, the second discriminator of R^2GAN makes prediction of modality source on the generated images rather than embeddings. The design enables R^2GAN to encapsulate a rich set of loss functions as well as using two-level ranking losses for effective learning of compatible features.

3. R^{2}GAN

3.1. Preliminaries

Problem Formulation. The goal of image-to-recipe retrieval is to search for relevant recipes that textually describe the preparation of a dish given a food image as query. Similar but in the reverse direction, recipe-to-image retrieval is to rank food images according to the likelihood of being cooked based on a given recipe. Denote $P = \{p_i = (r_i, v_i)\}_{i=1}^{N}$ as a set of N recipe-image pairs, where $r_i \in R$ is a recipe and $v_i \in V$ is its food image. The notations R and V denote the collections of recipes and images respectively. A pair p_i may be assigned a semantic label $c_i \in C$, where $C \in \mathbb{R}^k$ represents the set of k food categories such as waffle, spaghetti bolognese and chicken quesadilla, which correspond to the predefined food groups of recipes. It is worth noting that each image belongs to a unique recipe, while each recipe is allowed to contain more than one image. Furthermore, the state of an image is assumed "after cooking", meaning that an image captures only a fully prepared dish.

Due to the domain gap between recipe and image, the extracted raw features from both domains cannot be matched for similarity measurement. Similar in spirit as [35, 2], this paper aims to learn a common latent subspace to enable cross-modal comparison between recipe and food image. Specifically, a mapping function $\Psi(R, V) \rightarrow (\mathbf{E_R}, \mathbf{E_V})$ needs to be learnt. Given n recipe-image pairs, the function Ψ produces both recipe embeddings $\mathbf{E_R}$ and image embeddings $\mathbf{E_V}$, where $\mathbf{E_R} \in \mathbb{R}^{n \times d}$, $\mathbf{E_V} \in \mathbb{R}^{n \times d}$, and d is the dimension of the learnt embedding.

Generative Adversarial Network. The vanilla GAN [10] is composed of a generator G and a discriminator D which can be trained simultaneously in an adversarial way. The generator G is trained to capture the real data distribution p_{data} and generate fake images to fool discriminator D. On the other hand, the discriminator D is trained to distinguish between real and fake images. Specifically, G and D play a minmax game to optimize the following objective function:

$$\begin{aligned}\min_G \max_D V(D, G) =& \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \\ & \mathbb{E}_{z \sim p_z(z)}[\log (1 - D(G(z)))],\end{aligned} \tag{1}$$

where x is the real image with a data distribution p_{data}, and z is a noise with a prior distribution p_z.

3.2. Model Architecture

Figure 2 depicts the model architecture of our R^2GAN. The architecture is composed of two modules for recipe and

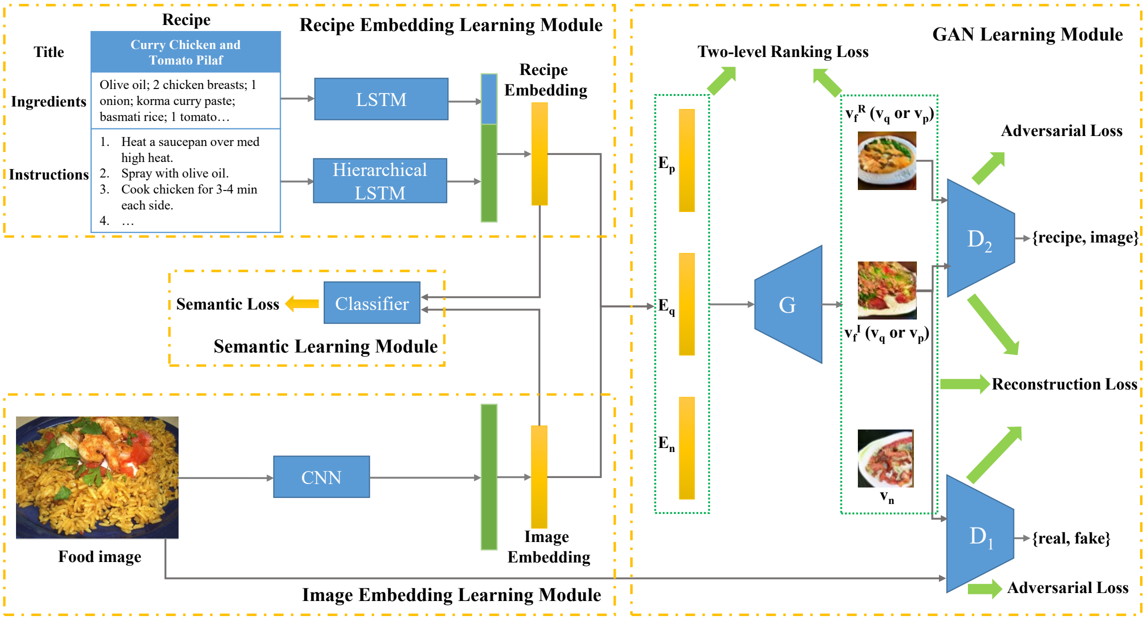


Figure 2. R^2GAN is composed of two modules for recipe and image embeddings and two modules for learning of GAN and semantic classification. The GAN learning module is redesigned with one generator (G) and two discriminators (D_1 and D_2) for cross-modal feature learning. Leveraging on the proposed GAN module, two-level ranking loss at embedding and image spaces is introduced.

image embeddings, and two modules for learning of GAN and semantic classification. The architecture is learned in an end-to-end fashion.

Recipe Embedding Learning. This module follows the work of [35], which employs a bi-directional LSTM and a hierarchical LSTM for representation learning of ingredients and cooking instructions respectively. The learnt representations are concatenated and fed into a fully connected layer for learning of recipe embedding.

Image Embedding Learning. Similar as other works in cross-modal recipe retrieval [35, 2, 5], the state-of-the-art ResNet-50 model is employed to extract image feature. We remove the last softmax classifier layer of ResNet-50 and initialize the rest layers with parameters pretrained in ImageNet ILSVRC12 dataset [33]. The resulting feature is further mapped by a fully connected layer to produce an image embedding in the same dimension as a recipe embedding.

GAN Learning. This module is specifically designed to learn compatible and explainable embeddings for image-recipe pairs. We redesigned vanilla GAN with one generator and two discriminators for cross-modal feature learning. As shown in Figure 2, the generator G is trained to be capable of reconstructing image from either recipe or image embedding. The reconstructed images from recipe and image embeddings are denoted as v_f^R and v_f^I respectively, where the subscript f represents a fake or reconstructed image and the superscript indicates the recipe or image source.

The first discriminator D_1, similar to traditional GAN, is to distinguish between real and fake images, i.e., v_{real} and v_f^I. The second discriminator D_2, in contrast, is to differentiate between v_f^R and v_f^I to tell the source of modality. The intuition of having D_2 is to nudge the distribution of v_f^R to be as similar or compatible as v_f^I which is learnt from the original image v_{real}. The generator G plays a special role in transforming textual recipe embeddings to images that are difficult for D_2 to predict the source. This min-max game played by GAN learning module novelly provides feedback to make the learnt recipe embedding self-explainable, specifically by having G to recount the visual appearance of an embedding for D_2 to make judgement. Note that this procedure naturally simulates an interpretable cross-modal retrieval, by showing user v_f^R as an explanation of how a recipe is visually interpreted and ranked by a system. In short, by having two discriminators, R^2GAN effectively enforces v_f^I to learn from real food image v_{real} and then v_f^R from v_f^I, until reaching a state where the reconstructed images from a different modality share similar or even a same distribution with the original image.

Semantic Learning. R^2GAN also takes advantage of high-level semantics (i.e., food categories) to assist the learning of recipe and image embeddings. Intuitively, both modalities should exhibit the same semantic interpretation when projected to the same common subspace.

3.3. Objective Formulation

Two-level Ranking Loss. Similar to other cross-modal retrieval methods [17, 38], triplet ranking loss is employed. Different from these works, nevertheless, R^2GAN considers two-level of losses due to embedding and reconstruction. Let E represent an embedding, v as a reconstructed image, and the subscripts q, p and n refer to query, positive and negative candidates respectively. We use a large-margin based ranking loss function which can be formalized as follows:

$$\begin{aligned} L_{rank} = &\max\{d(E_q, E_p) - d(E_q, E_n) + \alpha_1, 0\} + \\ &\mu \max\{d(v_q, v_p) - d(v_q, v_n) + \alpha_2, 0\}, \end{aligned} \tag{2}$$

where $d(\cdot, \cdot)$ is a distance function measuring the similarity between a given pair of query and candidate, for example, (E_q, E_p) as a positive embedding pair and (v_q, v_p) as the corresponding image pair. Note that the elements of a pair belong to different modalities. The parameters α_1 and α_2 are margins, and μ is a trade-off hyperparameter.

The two-level ranking loss enhances the robustness of learning, through enforcing the distances between positive pairs to be always smaller than negative pairs, not only in the embedding space but also the reconstructed image space. We use cosine similarity as distance function for embedding space as [35, 2], and pixel-wise Euclidean distance for image space.

Adversarial Loss. The three parts of R^2GAN, i.e., G, D_1, D_2, are optimized alternatively by adversarial training. Due to use of two discriminators, the losses produced by D_1 and D_2 are averaged as the training loss of G. Therefore, the GAN module losses are as follows:

$$\begin{aligned} L_{D_1} = &\mathbb{E}_{x \sim p_{image}}[\log D_1(x)] + \\ &\mathbb{E}_{\mathbf{E_V} \sim p_{image}}[\log(1 - D_1(G(\mathbf{E_V})))], \end{aligned} \tag{3}$$

$$\begin{aligned} L_{D_2} = &\mathbb{E}_{\mathbf{E_V} \sim p_{image}}[\log D_2(G(\mathbf{E_V}))] + \\ &\mathbb{E}_{\mathbf{E_R} \sim p_{recipe}}[\log(1 - D_2(G(\mathbf{E_R})))], \end{aligned} \tag{4}$$

$$\begin{aligned} L_G = &\frac{1}{2}(\mathbb{E}_{\mathbf{E_V} \sim p_{image}}[\log(1 - D_1(G(\mathbf{E_V})))]) + \\ &\mathbb{E}_{\mathbf{E_R} \sim p_{recipe}}[\log(1 - D_2(G(\mathbf{E_R})))], \end{aligned} \tag{5}$$

where $\mathbf{E_R}$ and $\mathbf{E_V}$ denote embeddings of recipe and image respectively.

Reconstruction Loss, which also considers two-level of losses in feature and image levels, is introduced to encourage the reconstructed images to retain as much as information of the original image. The reconstruction loss is defined as follows:

$$\begin{aligned} L_{recon} = &\frac{1}{2}(\left\|\Phi(v_{real}) - \Phi(v_f^I)\right\|_2^2 + \\ &\left\|\Phi(v_f^I) - \Phi(v_f^R)\right\|_2^2 + \\ &\beta(\|v_{real} - v_f^I\|_2^2 + \|v_f^I - v_f^R\|_2^2)), \end{aligned} \tag{6}$$

where $\Phi(\cdot)$ is a feature extractor for the input image, v_{real} stands for real food image, and the images v_f^I and v_f^R are reconstructed from image and recipe embeddings respectively. Following the practice in [7], the output before last layer of the discriminator is used as $\Phi(\cdot)$[1]. The term $\|\Phi(v_1) - \Phi(v_2)\|_2^2$ refers to feature-level loss and the term $\|v_1 - v_2\|_2^2$ refers to the image-level loss, with both using Euclidean distance. The parameter β controls the relative importance between feature and image losses.

Semantic Loss is characterized by cross-entropy loss as following:

$$L_{sem} = -\log \frac{\exp(E_c)}{\sum_i \exp(E_{c_i})}, \tag{7}$$

where E_c denotes either a recipe or image embedding category.

Overall Loss. The four modules of R^2GAN are learnt end-to-end. However, the parameters of modules are optimized separately using different loss functions. The full loss, defined as following, is used to update the parameters of embedding and semantic modules:

$$L_{full} = L_{rank} + \gamma L_{recon} + \lambda L_{sem}, \tag{8}$$

where γ and λ are trade-off hyperparameters.

On the other hand, the parameters of two discriminators are updated by L_{D_1} and L_{D_2}, while the parameters of generator G are updated by incorporating adversarial and reconstruction losses as following:

$$L_{G_{full}} = L_G + \delta L_{recon}, \tag{9}$$

where δ balances the relative importance of the two parts.

4. Experiments

4.1. Experiment Settings

Dataset. Recipe 1M [35] is the only large-scale food dataset with English recipes and images publicly available. The raw dataset contains more than 1 million recipes and almost 900,000 images. The experiments are conducted on the pre-processed recipe-image pairs provided by [35], which have totally 340,922 pairs with 70% for training, 15% for validation and 15% for testing. Each pair is assigned to one of the 1,048 semantic food categories compiled by [35].

Evaluation Metrics. Median rank (MedR) and recall rate at top K (R@K) are used to evaluate retrieval accuracy. MedR refers to the median rank position of true positives for all the testing queries. R@K measures the fraction of

[1] An alternative way of computing $\Phi(\cdot)$ is by using VGG network [14]. However, there is no obvious performance difference between these two approaches in our in-house experiment.

Size	Methods	image-to-recipe				recipe-to-image			
		MedR	R@1	R@5	R@10	MedR	R@1	R@5	R@10
1K	Random	500	0.1	0.5	1.0	500	0.1	0.5	1.0
	CCA [35]	15.7	14.0	32.0	43.0	24.8	9.0	24.0	35.0
	JNE [35]	5.2	24.0	51.0	65.0	5.1	25.0	52.0	65.0
	ATTEN [5]	4.6	25.6	53.7	66.9	4.6	25.7	53.9	67.1
	AdaMine [2]	2.5	36.4	66.2	76.9	2.1	37.4	66.7	77.1
	R^{2}GAN	**2.0**	**39.1**	**71.0**	**81.7**	**2.0**	**40.6**	**72.6**	**83.3**
10K	JNE [35]	41.9	-	-	-	39.2	-	-	-
	ATTEN [5]	39.8	7.2	19.2	27.6	38.1	7.0	19.4	27.8
	AdaMine [2]	16.5	12.5	31.5	42.2	15.6	13.6	32.8	43.4
	R^{2}GAN	**13.9**	**13.5**	**33.5**	**44.9**	**12.6**	**14.2**	**35.0**	**46.8**

Table 1. Cross-modal retrieval performance comparison in terms of MedR (median rank) and R@K (recall@K). A lower MedR and a higher R@K indicate a better model. The symbol "-" means that the results are not available in the original paper.

true positives being ranked at top K returned results. Therefore, a retrieval model with lower MedR and higher R@K is preferable.

Implementation. The output dimensions of ingredient and cooking instruction are set to 300 and 1,024 respectively. Meanwhile, the embeddings of both recipe and image are fixed to be in $d = 1024$ dimensions, following [35]. The design of the GAN learning module is guided by DCGAN [30]. The generator G consists of upsampling layers, each followed by batch normalization and ReLU activation except for the last layer which uses Tanh. We use the nearest-neighbor upsampling following a 3×3 stride 1 convolution as adopted by StackGAN [40]. For discriminator, strided convolution is adopted for down-sampling, with each followed by batch normalization and LeakyReLU activation except for the last layer which uses Sigmoid. Both discriminators D_1 and D_2 share the same architecture. The slope for LeakyReLU is set to be 0.2. As R^2GAN emphasizes more on embedding compatibility than image quality, the resolution of generated images is set to be 64×64 which is a typical size of thumbnail enough for visualization.

For all the experiments, Adam solver with adaptive learning schema [16, 2] is used with a batch size of 128. The initial learning rate of the R^2GAN is 0.0001 with a decay by multiplying 0.5 when the model reaches a plateau. The GAN learning module is trained with an initial learning rate of 0.0002, decaying by multiplying 0.1 every 20 epochs. During end-to-end training, with the principle that ranking loss is one order of magnitude bigger than other losses, we set μ=0.1 (Equation 2), β=1 (Equation 6), γ=0.01 and λ=0.01 (Equation 8). Following the usual practice in the literature, the margins α_1 and α_2 of two-level ranking loss in Equation 2 are set to be 0.3. The balance factor in Equation 9 is set to be δ=1 in order to balance adversarial and reconstruction loss.

The model training is conducted as following. In the first 20 epochs, the ResNet-50 weights are frozen and other parts of the model are trained from scratch. After that, we free the ResNet-50 weights and train the whole model for another 80 epochs. The strategy of triplet sampling is to generate samples from the mini-batch. Given a batch of matched image-recipe paires, if we choose one item from one modality as query E_q, then the corresponding item from another modality is treated as positive E_p while the rest are averaged as negative E_n. The three embeddings, i.e., the query and its positive and negative counterparts, are subsequently utilized as inputs for generator G to reconstruct images with corresponding outputs v_q, v_p and v_n (Equation 2 and Figure 2). Finally, the model with the best MedR performance on validation set is selected for testing.

4.2. Retrieval Results

Comparison. R^2GAN is compared against three state-of-the-art deep learning based approaches [35, 5, 2] and two baselines based on random and CCA [31]. Same as [35, 5], retrieval is conducted on a subset formed by random sampling of recipe-image pairs from the testing set. The recipe and image of a pair take turn as a query to retrieve its counterpart from the subset. The sampling process is repeated for 10 times and the mean retrieval results are reported. Note that, different from [2], the sampling process will not guarantee unique subsets without overlapping samples. In addition, when calculating MedR, the ranking position starts from 1 instead of 0, which is used by [35, 5]. In the experiment, we use the pretrained embeddings[2] provided by [2] and report their results on the subsets sampled by us.

Table 1 lists the performances of different approaches on 1K and 10K subsets. First, deep learning models significantly outperform all the baselines with large margin. Second, R^2GAN exhibits the best performance across all the evaluation measures among the deep models. Comparing to AdaMine [2] which reported the to-date best performance

[2] `https://github.com/Cadene/recipe1m.bootstrap.pytorch/tree/pytorch0.2#pretrained-models`

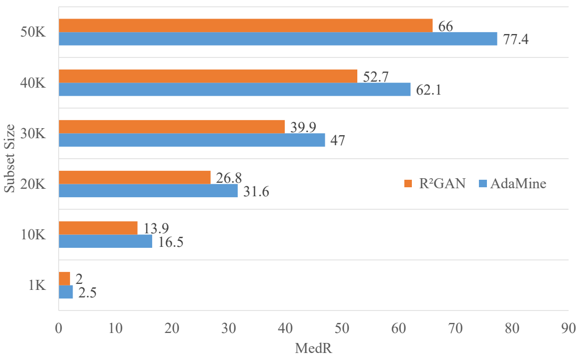
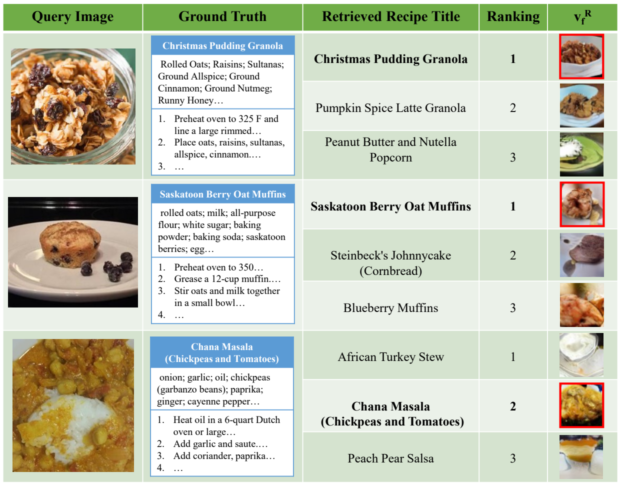


Figure 3. Scalability test between R^2GAN and AdaMine [2] for image-to-recipe retrieval.

on Recipe1M, R^2GAN manages to boost MedR by almost three ranking positions in both image-to-recipe and recipe-to-image retrieval in 10K setting. Observed from the similar thumbnails generated from image and recipe embeddings, we attribute the improvement to the peculiar design of the GAN learning module which enforces the embedding module to learn more compatible features.

Scalability. To investigate the robustness R^2GAN against large dataset beyond 10K, we further compare its MedR performance against AdaMine. For image-to-recipe retrieval, as shown in Figure 3, the gap between R^2GAN and AdaMine becomes obvious and larger with the increase of subset size. On the 50K dataset, which is almost equivalent to the original size of testing set provided by [35], R^2GAN manages to rank the true positive by 11.4 positions ahead of AdaMine on average, which is statistically significant. Similar results are also obtained for recipe-to-image search, where R^2GAN ranks true positives by 14 positions ahead on 50K dataset. Nevertheless, the MedR of R^2GAN, although much better than AdaMine, only reaches 66 for image-to-recipe retrieval in 50K setting, which shows the challenge of this task.

Visual Interpretability. The basic idea is to show thumbnails along each retrieved recipe such that user can browse through the search results quickly, while picking the right recipe even if it is not ranked at the top position. Figure 4 shows three typical examples of search in the experiment. In the first example (top), the ground-truth recipe is successfully ranked at the 1st place. The generated image is obviously more similar to query than others, demonstrating the interpretability of the generated images in explaining search results. In the second example (middle), both of the recipes ranked at 1st and 3rd positions belong to muffin. However, the image generated from ground-truth recipe has shape and layout more similar to query, which explains why it is ranked higher than other muffin recipes. In the third example (bottom), although the ground-truth recipe is ranked at the 2nd place, user may still pick this as result judging from the similarity of the generated image and query.



Query Image	Ground Truth	Retrieved Recipe Title	Ranking	v_f^R
	Christmas Pudding Granola Rolled Oats; Raisins; Sultanas; Ground Allspice; Ground Cinnamon; Ground Nutmeg; Runny Honey… 1. Preheat oven to 325 F and line a large rimmed… 2. Place oats, raisins, sultanas, allspice, cinnamon.… 3. …	**Christmas Pudding Granola**	**1**	
		Pumpkin Spice Latte Granola	2	
		Peanut Butter and Nutella Popcorn	3	
	Saskatoon Berry Oat Muffins rolled oats; milk; all-purpose flour; white sugar; baking powder; baking soda; saskatoon berries; egg… 1. Preheat oven to 350… 2. Grease a 12-cup muffin.… 3. Stir oats and milk together in a small bowl… 4. …	**Saskatoon Berry Oat Muffins**	**1**	
		Steinbeck's Johnnycake (Cornbread)	2	
		Blueberry Muffins	3	
	Chana Masala (Chickpeas and Tomatoes) onion; garlic; oil; chickpeas (garbanzo beans); paprika; ginger; cayenne pepper… 1. Heat oil in a 6-quart Dutch oven or large… 2. Add garlic and saute.… 3. Add coriander, paprika… 4. …	African Turkey Stew	1	
		Chana Masala (Chickpeas and Tomatoes)	**2**	
		Peach Pear Salsa	3	

Figure 4. Examples showing the interpretability of R^2GAN. By judging from the generated images (last column) from recipes, one can easily guess the ground-truth recipes of query images.

4.3. Ablation Studies

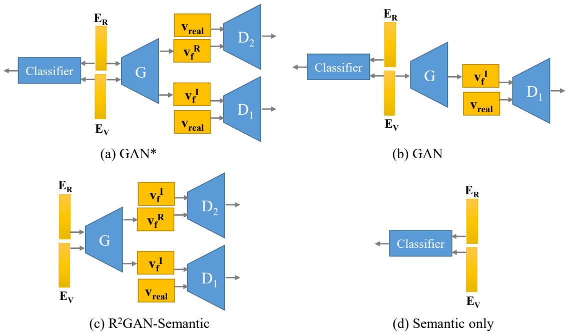
This section studies improvement due to different modules of R^2GAN. Figure 5 shows four variants of R^2GAN as following. To investigate the significance of Discriminator D_2, two variants, GAN* and GAN, are derived. Referring to Figure 5(a), GAN* modifies D_2 to guess between real image and the fake image constructed from a recipe, versus D_2 in R^2GAN which predicts the source of modality when an image is generated. GAN (see Figure 5(b)), on the other hand, simply removes D_2, which makes it equivalent to the original GAN except also considering semantic loss. As claimed in JNE [35] and ATTEN [5] that food semantics play an important role, we also study the performance of two other variants without semantic classification (i.e., R^2GAN-Semantic in Figure 5(c)) and with only semantic classification (i.e., Semantic only in Figure 5(d)). Additionally, we also compare to a variant, R^2GAN-, which employs conventional one-level ranking loss without image-level ranking loss. In other words, Equation 2 is modified as follows:

$$L_{rank} = \max\{d(E_q, E_p) - d(E_q, E_n) + \alpha_1, 0\}, \quad (10)$$

Table 2 lists the results of ablation study. First of all, the baseline GAN already outperforms all the previous models including AdaMine on this dataset. However, GAN*, which uses a variant of D_2, exhibits worse performance than GAN which is without D_2. The result is not surprising because reconstruction of image from recipe is highly difficult. Directly learning to imitate real image can re-

Methods	image-to-recipe					recipe-to-image				
	10K	20K	30K	40K	50K	10K	20K	30K	40K	50K
Semantic only	16.0	30.6	45.7	60.8	75.7	15.1	28.6	42.8	56.8	70.9
R^2GAN-Semantic	19.3	37.8	55.9	74.1	92.9	18.1	35.6	52.7	69.8	87.0
GAN	15.8	30.7	45.7	60.3	75.2	14.2	28.1	41.9	55.4	69.0
GAN*	19.3	37.9	56.1	74.2	92.9	17.2	34.0	50.5	67.1	83.4
R^2GAN-	14.6	28.4	42.0	55.2	69.0	13.2	25.2	37.5	49.9	61.9
R^2GAN	**13.9**	**26.8**	**39.9**	**52.7**	**66.0**	**12.6**	**24.2**	**35.7**	**47.4**	**59.0**

Table 2. Ablation study. Results are reported in terms of MedR with different subset sizes.



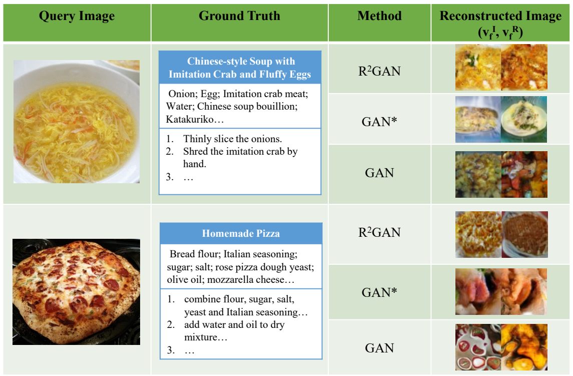
(a) GAN*

(b) GAN

(c) R^{2}GAN-Semantic

(d) Semantic only

Figure 5. Variants of architectures derived from R^2GAN for ablation study.



Query Image	Ground Truth	Method	Reconstructed Image (v_f^I, v_f^R)
	Chinese-style Soup with Imitation Crab and Fluffy Eggs Onion; Egg; Imitation crab meat; Water; Chinese soup bouillion; Katakuriko… 1. Thinly slice the onions. 2. Shred the imitation crab by hand. 3. …	R^{2}GAN	
		GAN*	
		GAN	
	Homemade Pizza Bread flour; Italian seasoning; sugar; salt; rose pizza dough yeast; olive oil; mozzarella cheese… 1. combine flour, sugar, salt, yeast and Italian seasoning… 2. add water and oil to dry mixture… 3. …	R^{2}GAN	
		GAN*	
		GAN	

Figure 6. Comparison of images generated by R^2GAN, GAN* and GAN. The last column shows the thumbnails reconstructed from image embedding v_f^I and recipe embedding v_f^R.

sult in overfitting harmful to the overall end-to-end learning. Instead, indirectly learning as in R^2GAN to imitate fake image generated from image embedding, which is inherently an easier task, appears to be more effective. The result listed in Table 2 also aligns with [35, 5] where semantic loss plays a critical role. Semantic-only, which is without GAN, performs better than its counterpart R^{2}GAN-Semantic, which is with GAN only but without semantics. The proposed R^2GAN successfully compromises both information, i.e., semantics and GAN, and shows the consistently best performances across subsets of different sizes from 10K to 50K. Comparing two-level versus one-level ranking loss, R^2GAN also shows incremental improvement over R^{2}GAN- consistently across all the subsets. Figure 6 compares the images generated from image and recipe embeddings by different GANs. R^2GAN manages to generate thumbnails substantially more realistic than other variants and are apparently more similar to the original images.

5. Conclusion

We have presented a new network architecture based on GAN for cross-modal recipe retrieval, which attains the new state-of-the-art performance on Recipe1M dataset. R^2GAN, particularly, exhibits robustness against large-size dataset and is more scalable compared to other models. Through the experiments, we attribute the improvement to the design of architecture which makes the learning of embedding compatible across text and visual modalities. This can be evidenced from the high similarity in food images despite being generated from different modalities. These generated images also greatly facilitate the self-explaining of search results. Using more advanced GANs [1, 25] and generating higher resolution images [40] may further improve performance and enhance search result interpretation. Through ablation studies, we show that the design of dual discriminators plays an important role in boosting the retrieval performance. Finally, despite that the two-level ranking loss boosts performance by a relatively small margin, the improvement is consistently noticed across different sizes of subsets. While encouraging, R^2GAN currently considers only image generation from recipe and not vice versa. With the release of new dataset, such as [12] which includes processing images for every step of cooking instructions, potentially recipe-from-image is a mission-possible task which worth further investigation.

Acknowledgement

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11203517).

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv preprint arXiv:1701.07875*, 2017.

[2] Micael Carvalho, Rémi Cadène, David Picard, Laure Soulier, Nicolas Thome, and Matthieu Cord. Cross-modal retrieval in the cooking context: Learning semantic text-image embeddings. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pages 35–44, New York, NY, USA, 2018. ACM.

[3] Jingjing Chen and Chong-Wah Ngo. Deep-based ingredient recognition for cooking recipe retrieval. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 32–41. ACM, 2016.

[4] Jingjing Chen, Chong-Wah Ngo, and Tat-Seng Chua. Cross-modal recipe retrieval with rich food attributes. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1771–1779. ACM, 2017.

[5] Jingjing Chen, Chong-Wah Ngo, Fuli Feng, and Tat-Seng Chua. Deep understanding of cooking procedure for cross-modal recipe retrieval. In *Proceedings of the 2018 ACM on Multimedia Conference*, MM '18, New York, NY, USA, 2018.

[6] Jingjing Chen, Lei Pang, and Chong-Wah Ngo. Cross-modal recipe retrieval: How to cook this dish? In Laurent Amsaleg, Gylfi ór Gumundsson, Cathal Gurrin, Björn ór Jónsson, and Shin'ichi Satoh, editors, *MultiMedia Modeling*, pages 588–600, Cham, 2017. Springer International Publishing.

[7] Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. In *Advances in Neural Information Processing Systems*, pages 658–666, 2016.

[8] David Elsweiler, Christoph Trattner, and Morgan Harvey. Exploiting food choice biases for healthier recipe recommendation. In *Proceedings of the 40th international acm sigir conference on research and development in information retrieval*, pages 575–584. ACM, 2017.

[9] Jill Freyne and Shlomo Berkovsky. Intelligent food planning: personalized recipe recommendation. In *Proceedings of the 15th international conference on Intelligent user interfaces*, pages 321–324. ACM, 2010.

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[11] Jiuxiang Gu, Jianfei Cai, Shafiq Joty, Li Niu, and Gang Wang. Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7181–7189, 2018.

[12] Jun Harashima, Yuichiro Someya, and Yohei Kikuta. Cookpad image dataset: An image collection as infrastructure for food research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1229–1232. ACM, 2017.

[13] Yoshifumi Ito, Wataru Shimoda, and Keiji Yanai. Food image generation using a large amount of food images with conditional gan: ramengan and recipegan. In *Proceedings of the Joint Workshop on Multimedia for Cooking and Eating Activities and Multimedia Assisted Dietary Management*, pages 71–74. ACM, 2018.

[14] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[15] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[17] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. *arXiv preprint arXiv:1803.08024*, 2018.

[18] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[19] Niki Martinel, Gian Luca Foresti, and Christian Micheloni. Wide-slice residual networks for food recognition. In *Applications of Computer Vision (WACV), 2018 IEEE Winter Conference on*, pages 567–576. IEEE, 2018.

[20] Niki Martinel, Claudio Piciarelli, Christian Micheloni, and Gian Luca Foresti. A structured committee for food recognition. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 92–100, 2015.

[21] Weiqing Min, Bing-Kun Bao, Shuhuan Mei, Yaohui Zhu, Yong Rui, and Shuqiang Jiang. You are what you eat: Exploring rich recipe information for cross-region food analysis. *IEEE Transactions on Multimedia*, 20(4):950–964, 2018.

[22] Weiqing Min, Shuqiang Jiang, Linhu Liu, Yong Rui, and Ramesh Jain. A survey on food computing. *arXiv preprint arXiv:1808.07202*, 2018.

[23] Weiqing Min, Shuqiang Jiang, Jitao Sang, Huayang Wang, Xinda Liu, and Luis Herranz. Being a supercook: Joint food attributes and multimodal content modeling for recipe retrieval and exploration. *IEEE Transactions on Multimedia*, 19(5):1100–1113, 2017.

[24] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[25] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[26] Tu Nguyen, Trung Le, Hung Vu, and Dinh Phung. Dual discriminator generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2670–2680, 2017.

[27] Ferda Ofli, Yusuf Aytar, Ingmar Weber, Raggi al Hammouri, and Antonio Torralba. Is saki# delicious?: The food perception gap on instagram and its relation to health. In *Proceedings of the 26th International Conference on World Wide Web*, pages 509–518. International World Wide Web Conferences Steering Committee, 2017.

[28] Yuxin Peng, Jinwei Qi, and Yuxin Yuan. Cm-gans: Cross-modal generative adversarial networks for common representation learning. *arXiv preprint arXiv:1710.05106*, 2017.

[29] Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Nikhil Rasiwasia, Gert RG Lanckriet, Roger Levy, and Nuno Vasconcelos. On the role of correlation and abstraction in cross-modal multimedia retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):521–535, 2014.

[30] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[31] 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 *Proceedings of the 18th ACM international conference on Multimedia*, pages 251–260. ACM, 2010.

[32] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–58, 2016.

[33] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[34] Sina Sajadmanesh, Sina Jafarzadeh, Seyed Ali Ossia, Hamid R Rabiee, Hamed Haddadi, Yelena Mejova, Mirco Musolesi, Emiliano De Cristofaro, and Gianluca Stringhini. Kissing cuisines: Exploring worldwide culinary habits on the web. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1013–1021. International World Wide Web Conferences Steering Committee, 2017.

[35] 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 *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3068–3076. IEEE, 2017.

[36] Lone Brinkmann Sørensen, Per Møller, A Flint, Magni Martens, and A Raben. Effect of sensory perception of foods on appetite and food intake: a review of studies on humans. *International journal of obesity*, 27(10):1152, 2003.

[37] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. Adversarial cross-modal retrieval. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 154–162. ACM, 2017.

[38] Kaiye Wang, Qiyue Yin, Wei Wang, Shu Wu, and Liang Wang. A comprehensive survey on cross-modal retrieval. *arXiv preprint arXiv:1607.06215*, 2016.

[39] Fuzheng Zhang, Nicholas Jing Yuan, Kai Zheng, Defu Lian, Xing Xie, and Yong Rui. Exploiting dining preference for restaurant recommendation. In *Proceedings of the 25th International Conference on World Wide Web*, pages 725–735. International World Wide Web Conferences Steering Committee, 2016.

[40] Han Zhang, Tao Xu, and Hongsheng Li. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5908–5916. IEEE, 2017.

[41] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251. IEEE, 2017.