

Where and What Am I Eating? Image-based Food Menu Recognition

Marc Bolaños^{1,2}[0000–0001–9838–1435], Marc Valdivia¹, and Petia Radeva^{1,2}[0000–0003–0047–5172]

¹ Universitat de Barcelona, Barcelona, Spain

² Computer Vision Center, Bellaterra, Spain

Abstract. Food has become a very important aspect of our social activities. Since social networks and websites like Yelp appeared, their users have started uploading photos of their meals to the Internet. This phenomenon opens a whole world of possibilities for developing models for applying food analysis and recognition on huge amounts of real-world data. A clear application could consist in applying image food recognition by using the menu of the restaurants. Our model, based on Convolutional Neural Networks and Recurrent Neural Networks, is able to learn a language model that generalizes on never seen dish names without the need of re-training it. According to the Ranking Loss metric, the results obtained by the model improve the baseline by a 15%.

Keywords: multimodal learning · computer vision · food recognition

1 Introduction

Food and nutrition is one of the main activities in people’s lives. Nowadays, food does not only cover a basic need, but it has become a really important aspect of our social life. Since social networks appeared and, with them, food-focused applications (like TripAdvisor, Yelp, etc.), their users have started uploading photos of their meals to the Internet. It seems to be a strong and visible tendency in today’s society to share pictures of absolutely every piece of food that we taste; exotic or local, fancy-looking or ordinary. Moreover, people post, on many different social media channels, plenty of videos of special restaurants where they eat. Every single day, thousands of people use social media to make recommendations, promote a particular place or give their friends a warning about a nearby restaurant. That is why, tags and location opportunities were introduced for all social media users to make their posts easier and faster to create. The creation of automatic tools for food recognition based on images could enable an easier generation of content, create food diaries for improving nutrition habits or even create personal food profiles for offering personalized recommendations.

The purpose of this work is to explore a problem that we call image-based food menu recognition, which consists in, given an image, determine its correct menu item corresponding to the restaurant where it was taken (see Fig. 1). By



Fig. 1. Example of the Food Menu Recognition problem, where we have to retrieve the correct food name from a list of menu items.

being able to match the picture to an item of the menu it would be easier to retrieve the exact nutritional information of the food or any other data stored by the restaurant owners. Some of the main applications for this model would be creating a personalised profile with food preferences or a personal food diary for improving the eating habits.

The proposed methodology does not need to train a new model for each restaurant [32], instead it will learn to understand meal names in relation to a set of examples by learning a language model. We should point out the difficulty of the problem because of the context where we are working in. Restaurants usually use fancy names to refer to the dishes just to get the attention of their customers. Additionally, food presentation is different in every restaurant, having a high intra-class variability. Chefs try to surprise the customers by using unusual combinations of ingredients, colorful plates and/or sauces.

1.1 Health and Leisure

The work in [22] introduces the relationship that exists between food consumption and people’s health. In Europe, despite being a first-world region, more than 4 million people die each year due to chronic diseases linked to unhealthy lifestyles. In many of these cases, the lack of basic knowledge or awareness is a crucial factor in all problems, most people simply do not pay much attention to their eating habits. Furthermore, as it is mentioned in [24], a great number of deaths related to coronary heart diseases are caused by a group of major risk factors among which bad eating habits are at the top.

On the other hand, for a lot of people being and feeling healthy is considered a must. Thanks to social networks, people share their healthy lifestyle on social media on a daily basis. Nowadays, going out for dinner and enjoying a cosy atmosphere in a restaurant is not enough. The healthier (and better looking) your food is, the better. Because of this important fact, today’s restaurants are

really visible online and they tend to use many different Internet channels to remain in the center of their customers' attention. Food-based applications like Yelp, help their users find opinions on the quality of the service in the place they plan to visit, and all the data introduced is generated by the users with their smartphones.

1.2 Food Analysis and Deep Learning

Considering the huge number of pictures of meals that people upload on the Internet, food analysis has become popular in the Deep Learning field. That is the reason why several public datasets have appeared. Some examples of public well-known datasets are Food-101 [8], UEC Food256 [16], or Vireo-Food 172 [9]. The most basic problem related to food explored in the literature is food detection [1], which consists in determining if any kind of food appears in an image. Food recognition is one of the most popular problems nowadays [2]. It consists in recognizing the food present on a picture given a pre-defined set of classes (dishes). Other applications of food analysis are food localization, which consists in detecting multiple dishes in a picture [7], calories estimation [13], ingredients detection [5], or multi-dishes recognition for self-service restaurants [3], which combines several of the aforementioned problems.

1.3 Restaurant Food Recognition

Several applications are focused on understanding customers' experiences in restaurants. Some sites like Yelp have plenty of information, but they are not able to classify a picture in the restaurant's menu automatically. It is the user who must do this manually. For this reason, we propose a model to solve this specific problem: locate the restaurant where customers are eating and recognize the meal that they chose from the menu [32]. Solving this problem would allow to create automatic personalized food diaries or personal food preferences, among other applications. The novelties of our work are the following:

- We propose a model that determines the similarity between a picture of food and the dish name provided in the restaurant's menu. Thanks to the language model learned, the system is able to detect the most probable food item in the menu using semantic information from LogMeal's API.
- We propose the first model for food menu recognition applicable to any restaurant. The system does not need previous information of a specific restaurant or a set of examples for a specific class to perform the prediction.
- We make public a dataset collected from Yelp³. Our dataset contains 53,877 images, from 313 restaurants and 3,498 different dishes.
- The results obtained over the collected data improve the baseline by a 15%.

In the context of the dataset, although ours is equivalent to the one proposed in [32], which is in Chinese, we were not able to perform tests on their dataset

³ <http://www.yelp.com>

due to language issues. A critical component of our methodology is the language model, which allows to generalize for any restaurant, but considering the lack of embedding models pre-trained on Chinese, it is not possible to directly apply it.

This paper is organized as follows. In the related work (see Sec. 2), we explain previous papers published in relation with the problem that we want to solve. Our proposed model is introduced in the methodology (see Sec. 3). The dataset section (see Sec. 4) introduces the data used to train our model and how it was collected. In results (see Sec. 5), we explain and discuss the set of experiments done to choose the best parameters of the proposed model and their performance. Finally, we draw some conclusions and future work (see Sec. 6).

2 Related Work

Deep learning and Convolutional Neural Networks (CNNs) [17] have played a major role in the development of food-related methods in the last years. The huge amount of images related to food available on the internet in websites like Google Images, Instagram or Pinterest have allowed to collect large-scale datasets useful for training deep learning architectures. Even though, challenges inherent to the culinary world like intra-class variability (e.g. apple pie) and inter-class similarity (e.g. different types of pasta), demand the use of complex and smart algorithms. In this section we review the literature on works related to food analysis problems, some important works on multi-modal learning and food, and the application of these techniques in the restaurants context.

2.1 Food Analysis

In the literature there exist several problems and topics related to the analysis of food images. One of the most notable topics is food detection [1, 23], where the goal is to detect whether a given image contains any food-related information/element. In a similar way, food recognition [2, 20, 25] is a widely explored topic, being the goal in this case to classify the image into a set of pre-defined list of classes related to food (usually prepared meals).

Other problems explored in the literature that are related to food analysis are calorie counting and monitoring or volume estimation, like in [31, 19], where the authors present a mobile phone-based calories monitoring system to track the calories consumption for the users. Or focused on diabetes, Li et al. [18] estimate the amount of carbohydrate present in a meal from an image.

Other works have treated problems like food localization. In [7] the authors introduce the use of egocentric images to perform food detection and recognition. Food ingredients recognition [9, 5] uses a state of the art CNN to predict a list of ingredients appearing in the meal. Food localization and recognition on self-service restaurants is presented in [3].

2.2 Multi-modal Food Analysis

Some times, food analysis uses context or additional information to improve the accuracy of the predictions. This complementary data can be of several types (e.g. images or text). Multi-modal Deep Learning [21] solves this particular problem, learning features over multiple modalities. The paper in [26] introduces a new large-scale dataset with more than 800.000 images and 1.000 recipes. The predictive model presented in the paper tries to join images and recipes through a retrieval task. The proposed solution generates two vectors. One of the vectors represents the image and the other one represents the recipe (text). For optimizing the model, they use the cosine similarity loss, which determines if a given recipe-image pair represents the same food.

The problem that we face also has two different inputs: we need to compare an image and a text sequence, so it could also be formulated as an image retrieval problem. The main difference of our proposal is that, instead of using a general purpose CNN to generate the features vector of the image, we use a semantic-based system for generating food categories that will be structured as a feature vector. Additionally, we use the dish name (text) instead of the recipe and intend to classify the input image into a set of menu items, being a problem more related to restaurant food recognition.

2.3 Restaurant Food Recognition

Seeing food analysis from a different perspective, in [4] the authors propose an automatic food logging system using smartphones. They use state of the art computer vision techniques and add context information of the restaurant to predict the food being consumed by the customer. The system in [31] creates a calorie estimation from web video cameras in fast food restaurants across the United States. They focused on a reduced group of restaurants to understand the obesity problem. Similarly to our proposal, Xu et al. [32] introduces the context of the pictures to recognize the dish appearing in the image. Using the GPS information provided by the smart-phones they can determine a set of possible restaurants where the picture has been taken. This reduces the search space, which is really important when you try to determine the restaurant and menu item that appear in the picture taken by the user.

The system in [32] needs to train a discriminative model for each pair of restaurants in the dataset comparing their menus and images. Another common problem present in food recognition (or object recognition in general) is that it is limited to a predefined set of classes. This means that if the model was not trained to recognize a specific type of food, it will never provide it as a possible output. Furthermore, the complexity in the restaurants' food recognition resides in the need of training a different model for each restaurant. These models could be very accurate, but the number of outputs is also limited to the restaurant's menu. In this paper, we propose a model that solves these problems. It learns a language model considering a great amount of possible names and associates them to their corresponding pictures. Thus, our algorithm should be able to

take a completely new restaurant’s menu (never seen before) and a totally new picture associated to one of the menu’s items and find out the correct menu item given the list. Thus, implying that the proposed model does not need to specifically learn every meal.

3 Image-based Food Menu Recognition: Our Model

Fig. 2 shows a scheme of our proposed model, which is based on image retrieval. Given two inputs: an image, and a dish name, it gives an output value based on their similarity. By using this, the prediction process consists in running the predictive model for each menu item and a single meal picture. The generated results produce a ranked list based on the most-similar-first criterion.

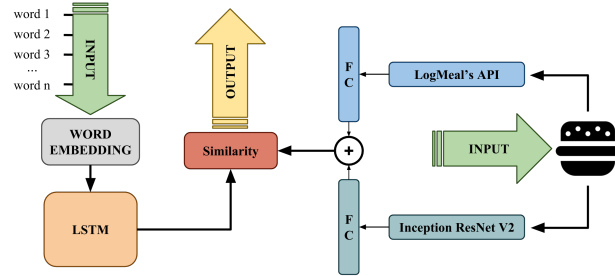


Fig. 2. Image-based food menu recognition model. On the one hand, the system gets an image and applies two different CNNs to generate the feature vectors. Each one is connected to a different fully connected layer to generate comparable structures and are combined performing an addition operation. On the other hand, the text sequence is processed by a word embedding and a Long Short Term Memory. Finally, we compute the similarity between the two inputs using the Euclidean similarity.

3.1 Image and dish name embedding

Our method takes two different inputs, one in the form of an image, that will be transformed in two vectors of different modalities: a low-level vector and a high-level semantic vector, and the other in the form of text. Which means that they must be treated differently before embedding them into the system.

First, the image is converted in two vectors in parallel. One of them provides a low-level description of the food image by using the penultimate layer of the InceptionResNetV2 [29] CNN, composed by a vector of 1,536 values. This CNN is pre-built in the Keras [10] framework and trained using the ImageNet [12] dataset. The other vector provides a high-level semantic description of the food

appearing in the image by using LogMeal’s API⁴. This API provides three different CNNs that predict the dish [2], food group (or family) [2] and the ingredients detected in the image [5]. More precisely, LogMeal’s API provides (during the development of this paper) as output the probabilities of the image of belonging to 11 food groups (e.g. meat, vegetables, fish, soup, etc.), 200 dishes (e.g. pizza, spaghetti alla carbonara, etc.) and 1.092 ingredients (e.g. tomato, cheese, salt, garlic, etc.). In the implementation of our model, we are not using the ingredients output because, as we observed, the large dimensionality of the output and the noise that this group introduces to the system does not help obtaining better results. This, in order to build the semantic high-level vector, we concatenate the probabilities vector of the food groups together with the probabilities vector of the dishes.

Second, the text sequence input representing the meal’s name is encoded using a word embedding. The inputs of our dataset are, in most of the cases, in English or Spanish. For this reason, and in order to make our model converge quicker, we need a word2vector pre-trained system supporting multiple languages. This is why we chose ConceptNet [28], which generates vectors of 300 features. The words that do not appear in ConceptNet’s vocabulary are initialized using a vector of random values.

Unlike the two vectors extracted from the images, which are pre-computed and used as inputs to our system, the word embedding matrix is considered in the optimization procedure and trained together with the rest of the model.

3.2 Model structure

More details about the image feature vectors generation and embedding can be seen in Fig. 3. One of them comes from LogMeal’s API response and the other from the InceptionResNetV2. Later, each of them is inputted to the system and linked to a fully connected (FC) layer of 300 neurons. This layer transforms the feature vectors to the same size, so we can combine them applying an addition operation, which has been proven to be a simple yet effective way of multi-modal information merging [6].

Considering the text sequence that encodes the meal’s name, it is generated using a Long Short Term Memory (LSTM) [15] network (Fig. 4) that encodes and joins the sequence of word embedding vectors generated in the first step. In order to match the dimensions of the image vector, the output size of the LSTM is also set to 300 neurons.

3.3 Similarity and Ranking

The last part of the model consists in processing the vectors provided from the image side and the text side in order to calculate their similarity, which will be a value between 0 and 1. Nevertheless, given a certain image and all the list of items in a restaurant’s menu, we use the generated similarity values in order

⁴ <http://www.logmeal.ml>

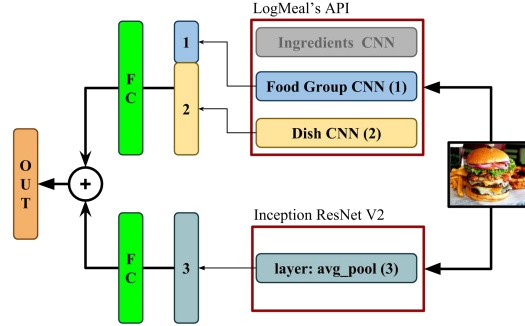


Fig. 3. Image processing part of our model. The system uses the food group and dish recognition outputs of LogMeal’s API to create a semantic vector and connect it to a FC layer. The penultimate layer of the InceptionResNetV2 CNN is also used in parallel as a low-level feature vector which is connected to another FC layer. Finally, both partial results are combined performing an addition.

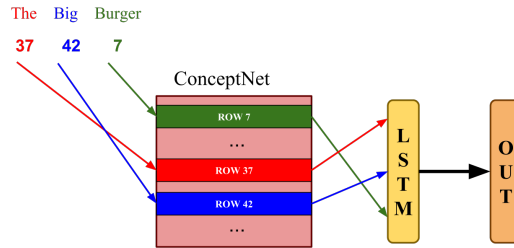


Fig. 4. The text sequence is encoded using a Word Embedding matrix, which is initialized using ConceptNet [28]. The generated vectors are connected to an LSTM.

to build a sorted ranked list. It means that we need to run the model for each item in the menu on the same picture. The similarity function used to build the algorithm is an adaptation of the Euclidean distance $\frac{1}{1+\|q-p\|}$.

4 Dataset

The dataset presented in this work was built using Yelp as the source of the information. We scraped the available public information of each restaurant, which consists in a list of menus for each restaurant, a list of dishes for each menu, and a list of images for each dish.

4.1 Dataset characteristics

The dataset was built from restaurants located in California. We chose this location because of the amount of active Yelp users in this area. We make the dataset publicly available⁵.

Analyzing the response of LogMeal’s API, we decided to remove the ingredients information. Analyzing the outputs for images of the same dish name, we observe that they have similar activation points, and at the same time they are different for images that represent different meals. Nevertheless, the ingredients recognition is noisy and does not give enough relevant information. Leading to an increase in the dimensionality of the input and a decrease of performance.

Table 1 (right) shows the number of images, dishes and restaurants in the dataset. The dataset dishes’ vocabulary is composed of 1,584 different words. Fig. 5 shows an histogram of the number of dishes per restaurant (left), and the number of images per dish (right). Observing the figures, the number of restaurants with just only one dish in their menu is considerably high, that is because we only retrieve the dishes containing some image. Additional problems that we found during dataset collection include the language of the dishes. Due to the location of the restaurants, there is a high probability of finding dishes in both English and Spanish, which introduces a problem: special characters. We encoded the text using the UTF-8 format, but there are some cases where the characters were represented by an empty symbol (-). We decided to remove these samples from the dataset in order to avoid errors during the word embedding.

4.2 Dataset Split

The dataset is split in three groups: training, validation and testing. Previously to the split process, we cleaned the data. This means removing the dishes encoded in a not valid format or the ones that do not have more than 5 images. The dishes are randomly split into three groups: the training group contains 80% of the dishes, 8% is included in validation and 12% of the meals are in the testing split. The number of images of the groups are shown in Table 1 (left).

⁵ Available after paper publication due to blind review process.

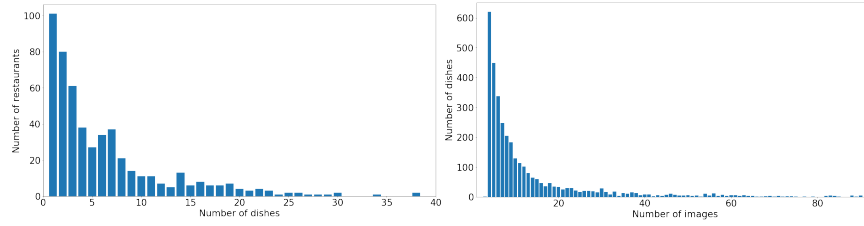


Fig. 5. Histogram of the number of dishes (with images) per restaurant in the dataset (left). Histogram of the number of images per dish in the dataset (right).

Table 1. Number of images in each split of the dataset (left). Number of images, dishes and restaurants of the dataset (right).

Split	# of images	Split	# of samples
Training	37,956	# of images	53,877
Validation	7,721	# of dishes	3,498
Test	10,794	# of restaurants	313

Considering that our model encodes the similarity of the image and text inputs, we need to provide both positive and negative samples in order to train it. The information downloaded from Yelp only contains positive examples, for this reason a set of negative samples has been generated for training (becoming a 50% of the total training samples). The negative examples have been generated assigning a wrong dish name to every image of the dataset. The validation and test splits are built randomizing the set of selected dishes in the menu together with the correct one. The groups of dishes were formed by randomly selecting between 10 and 20 dishes per menu. We generate a random list instead of using the menus of the restaurants to avoid restaurants that have few dishes in their menus.

5 Results

In this chapter we present the results obtained in our work, introduce the metrics used to evaluate the system and show the set of experiments created to find the best combination configuration of our model.

5.1 Ranking Loss & Accuracy Top-1 Distance

In order to compare the performance of the different methods, we use the Ranking Loss [30]. The lower the ranking loss is, the closer is the right value to the top of the list.

To complement the ranking loss error metric, we introduce our own accuracy metric in Eq. 1, which we call accuracy top-1 distance. This measure evaluates how close the ranked result is to the top. The difference with the ranking loss

is that our metric only takes in consideration the distance from the position of the predicted class to the top of the ranking. We normalize the output between 0 and 1 using the number of labels in our ranking.

$$\text{accuracy top-1 distance} = \frac{n_{\text{labels}} - 1 - \text{ranking}_{\text{position}}}{n_{\text{labels}} - 1} \quad (1)$$

5.2 Experimental setup

There are several components of our methodology that need to be tuned for finding the best configuration. The selection of the best combination of components was done using a forward propagation-grid search, and the policy we follow to choose the best parameter uses the ranking loss error over the test. The configurations to test were grouped in *similarity measures*, *losses*, *CNN features* and *sample weight*. For each step in the grid search, we select the configuration that obtains the best performance for each of the groups. Each configuration was calculated training the model 5 times. The representative model for each configuration was chosen considering the median value of the 5 runs. The results of the best configuration were obtained at the first epoch with a batch size of 64 samples and without applying any data augmentation or normalization process. Following, we detail the different model variants that we compare in the experimental section.

Similarity measures: We tested two similarity function candidates. a) the *Euclidean* similarity, which consists on a normalized version of the euclidean distance; and b) the *Pearson similarity* (see Eq. 2), which is the absolute value of the Pearson correlation. Using the absolute value we get values between 0 and 1.

$$\rho = \left| \frac{\text{cov}(X, Y)}{\sigma_x \sigma_y} \right| \quad (2)$$

Losses: We tested: a) the binary cross-entropy (BCE) [27], which is a commonly used loss function for binary classification problems; and b) the contrastive loss (CL) [14], which is usually used for Siamese networks [11]. The contrastive loss is a distance-based system and tries to minimize the separation between examples of the same semantic topic.

CNN features: We also tested different CNN feature extraction configurations in our model: a) using only the features from LogMeal’s API (*LM*); b) combination of the vectors from LogMeal and the InceptionResNetV2 CNN (*LM+Inc*); and c) InceptionResNetV2 only (*Inc*).

Sample Weight: The last configuration to test is the sample weight. It indicates whether we want to assign a weight value to each dish in relation with the amount of images that it contains with respect to the total number of images in the dataset. This kind of weighing is usually useful when the dataset is unbalanced, giving more importance to the samples that are less frequent.

5.3 Experimental Results

Table 2 shows the results of the grid search. The last row of the table displays the baseline error (based on a random selection of an item in the menu) and accuracy value over validation and test. We have to consider that the values of the ranking loss follow the rule, the lower the better. Meanwhile, the accuracy has the opposite behavior, we want to achieve the higher possible value. The first two rows of the table compare the two similarity measures. Both similarity measures are tested with the same loss optimizer, CNN and sample weight values to be comparable. The error of the Euclidean similarity is 0.033 points better than the one using the Pearson function. Comparing the loss functions, we can see that even though the contrastive loss is usually used for similarity-based CNN models, in this case the binary one works better. If we compare the different CNN feature extraction methods, LM and Inc, the first one works better. It is because LogMeal’s models are trained using food images. Despite this considerations, the best results are obtained by the model using the combination of the two CNNs, meaning that both networks complement each other. Finally, we see that we obtain better results if we deactivate the sample weights. The cause for this might be that we do not have a set of pre-established classes, but instead we have a language model that links them semantically. This component of our architecture is able to better learn the importance of each sample without the need of forcing a specific weight during optimization. Concluding the table analysis, the best combination of parameters for our model improves the baseline by a 15%. The best ranking loss for the test group is 0.351 and the accuracy top-1 distance is 0.678. It means an improvement of 0.149 and 0.178 points respectively over the baseline.

5.4 Visual Results Analysis

In Fig. 6 and 7 we show some visualizations of the results obtained by our model. The visualization contains a picture of the meal, the ranked results of our system and the true prediction for the image. Fig. 6 shows that the cases where the system works better is when the picture presents a single piece of food and the image is clear and centered as well as contains a common dish (with enough samples in the training set). Fig. 7 shows examples of failure cases, where the images contain multiple meals on them, making the recognition harder. Additionally, it is appreciable that the dishes with long names are usually at the bottom of the ranking. It is because these meals do not contain a lot of images and are not very popular in the restaurants. So, the model is not able to learn them and retrieve good predictions.

Another problem that we encountered was that, even being uncommon, the data tagged by Yelp’s users is misclassified because the pictures uploaded to the site are not verified. Sometimes, the users take photos of their dishes including context information, and it is a possibility that this information includes other people’s meals, which makes more difficult to classify the sample. The main difficulty for the algorithm is dealing with a high variety of names. The restaurants

Table 2. Comparison of results for the different model configurations. CNN feat. indicates the combination of CNNs used in the model (LogMeal’s API and InceptionResNetV2). The weight column indicates if the systems is using sample weight or not. The ranking loss is indicated with *r.loss* (the lower the better), and the accuracy top-1 distance is *acc.* (the higher the better). For each vertical section, a different configuration is tested. When a certain configuration is fixed it is shown in boldface.

similarities	losses	CNN feat.	weight	val		test	
				r. loss	acc.	r. loss	acc.
euclidean	binary	LM	NO	0.384	0.623	0.362	0.671
pearson	binary	LM	NO	0.416	0.602	0.395	0.639
euclidean	binary	LM	NO	0.384	0.623	0.362	0.671
euclidean	contrastive	LM	NO	0.405	0.398	0.375	0.664
euclidean	binary	LM	NO	0.384	0.623	0.362	0.671
euclidean	binary	LM+Inc	NO	0.372	0.641	0.351	0.678
euclidean	binary	Inc	NO	0.443	0.572	0.413	0.598
euclidean	binary	LM+Inc	NO	0.372	0.641	0.351	0.678
euclidean	binary	LM+Inc	YES	0.396	0.612	0.378	0.668
euclidean	binary	LM+Inc	NO	0.372	0.641	0.351	0.678
random selection (baseline)				0.5	0.5	0.5	0.5

have some speciality dishes that they name at their own. These meals are really difficult to classify, even for a human. Visualizing the results and analyzing the responses of a random selection of the predictions, we have found some properties that usually work better in our system. The meals that contain common food names tend to get better results than the ones with exotic names. This fact is due to two main reasons: the first one is that the dataset has a lot of examples with common names and can learn them better, and the second one is that the exotic names do not tend to appear at the word embedding matrix, so the system has no initial information of them. Moreover, these names are present in just a few restaurants, so the system does not have enough examples to learn from.

6 Conclusions and Future Work

We can conclude that it is possible to build a model for food restaurant menu recognition that generalizes for any restaurant available, without the need of learning a different model per restaurant or restaurant pairs. This result is achieved thanks to learning a language model that jointly embeds the information from all the dishes available together with low and high-level (semantic) information coming from the images. The contributions that we have done to the scientific community are the following:

- We introduce the use of a language model for dishes and semantic image information by means of LogMeal’s API to perform menu items recognition from restaurants.

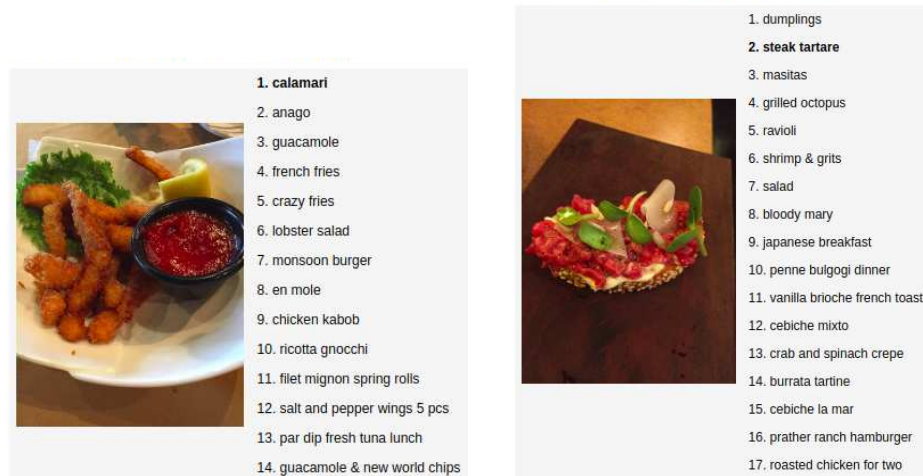


Fig. 6. Examples of ranked lists produced by our algorithm for images of the dishes 'calamari' (left) and 'steak tartare' (right). We observe the good results obtained when the names of the dishes are common enough.

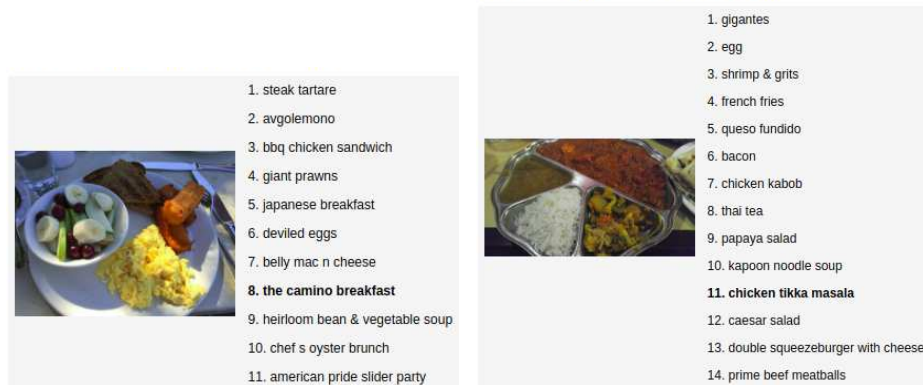


Fig. 7. Examples of ranked lists produced by our algorithm for images of the dishes 'the camino breakfast' (left) and 'chicken tikka masala' (right). Worse results are obtained when the names of the dishes are rare.

- We propose a new model that determines the similarity between a food image and a menu item of a restaurant without the need of re-training for each restaurant, which improves the baseline by a 15%.
- We present a new dataset composed by the dishes and images of the restaurant’s menu collected from Yelp. The dataset contains 53,877 images, 3,498 dishes and 313 restaurants.

One of the main issues to take into consideration in the future is the treatment of dishes with exotic names, which can not be easily learned by our language model. Furthermore, in the future we plan to introduce the GPS information of the images. The location of the user gives us a list of two or three candidate restaurants where they are eating. Combining the menus of these restaurants and applying the proposed system we would be able to determine where and what a person is eating.

References

1. Aguilar, E., Bolanos, M., Radeva, P.: Exploring food detection using cnns. arXiv preprint arXiv:1709.04800 (2017)
2. Aguilar, E., Bolaños, M., Radeva, P.: Food recognition using fusion of classifiers based on cnns. In: International Conference on Image Analysis and Processing. pp. 213–224. Springer (2017)
3. Aguilar, E., Remeseiro, B., Bolaños, M., Radeva, P.: Grab, pay and eat: Semantic food detection for smart restaurants. arXiv preprint arXiv:1711.05128 (2017)
4. Bettadapura, V., Thomaz, E., Parnami, A., Abowd, G.D., Essa, I.: Leveraging context to support automated food recognition in restaurants. In: Applications of Computer Vision (WACV), 2015 IEEE Winter Conference on. pp. 580–587. IEEE (2015)
5. Bolaños, M., Ferrà, A., Radeva, P.: Food ingredients recognition through multi-label learning. In: International Conference on Image Analysis and Processing. pp. 394–402. Springer (2017)
6. Bolaños, M., Peris, Á., Casacuberta, F., Radeva, P.: Vibiknet: Visual bidirectional kernelized network for visual question answering. In: Iberian Conference on Pattern Recognition and Image Analysis. pp. 372–380. Springer (2017)
7. Bolanos, M., Radeva, P.: Simultaneous food localization and recognition. In: Pattern Recognition (ICPR), 2016 23rd International Conference on. pp. 3140–3145. IEEE (2016)
8. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101—mining discriminative components with random forests. In: European Conference on Computer Vision. pp. 446–461. Springer (2014)
9. Chen, J., Ngo, C.W.: Deep-based ingredient recognition for cooking recipe retrieval. In: Proceedings of the 2016 ACM on Multimedia Conference. pp. 32–41. ACM (2016)
10. Chollet, F., et al.: Keras. <https://keras.io> (2015)
11. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. vol. 1, pp. 539–546. IEEE (2005)
12. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09 (2009)
13. Ege, T., Yanai, K.: Simultaneous estimation of food categories and calories with multi-task cnn. In: Machine Vision Applications (MVA), 2017 Fifteenth IAPR International Conference on. pp. 198–201. IEEE (2017)
14. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: Computer vision and pattern recognition, 2006 IEEE computer society conference on. vol. 2, pp. 1735–1742. IEEE (2006)
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural computation* **9**(8), 1735–1780 (1997)
16. Kawano, Y., Yanai, K.: Automatic expansion of a food image dataset leveraging existing categories with domain adaptation. In: European Conference on Computer Vision. pp. 3–17. Springer (2014)
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)

18. Li, H.C., Ko, W.M.: Automated food ontology construction mechanism for diabetes diet care. In: Machine Learning and Cybernetics, 2007 International Conference on. vol. 5, pp. 2953–2958. IEEE (2007)
19. Liu, C., Cao, Y., Luo, Y., Chen, G., Vokkarane, V., Ma, Y.: Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment. In: International Conference on Smart Homes and Health Telematics. pp. 37–48. Springer (2016)
20. Martinel, N., Foresti, G.L., Micheloni, C.: Wide-slice residual networks for food recognition. arXiv preprint arXiv:1612.06543 (2016)
21. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: Proceedings of the 28th international conference on machine learning (ICML-11). pp. 689–696 (2011)
22. Organization, W.H., et al.: Food and health in Europe: a new basis for action. World Health Organization. Regional Office for Europe (2004)
23. Ragusa, F., Tomaselli, V., Furnari, A., Battiato, S., Farinella, G.M.: Food vs non-food classification. In: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management. pp. 77–81. ACM (2016)
24. Rozin, P., Fischler, C., Imada, S., Sarubin, A., Wrzesniewski, A.: Attitudes to food and the role of food in life in the usa, japan, flemish belgium and france: Possible implications for the diet–health debate. *Appetite* **33**(2), 163–180 (1999)
25. Salvador, A., Hynes, N., Aytar, Y., Marin, J., Ofli, F., Weber, I., Torralba, A.: Learning cross-modal embeddings for cooking recipes and food images. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017)
26. Salvador, A., Hynes, N., Aytar, Y., Marin, J., Ofli, F., Weber, I., Torralba, A.: Learning cross-modal embeddings for cooking recipes and food images. *Training* **720**, 619–508 (2017)
27. Shore, J., Johnson, R.: Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy. *IEEE Transactions on information theory* **26**(1), 26–37 (1980)
28. Speer, R., Lowry-Duda, J.: Conceptnet at semeval-2017 task 2: Extending word embeddings with multilingual relational knowledge. arXiv preprint arXiv:1704.03560 (2017)
29. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: AAAI. vol. 4, p. 12 (2017)
30. Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining multi-label data. In: Data mining and knowledge discovery handbook, pp. 667–685. Springer (2009)
31. Wu, W., Yang, J.: Fast food recognition from videos of eating for calorie estimation. In: Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on. pp. 1210–1213. IEEE (2009)
32. Xu, R., Herranz, L., Jiang, S., Wang, S., Song, X., Jain, R.: Geolocalized modeling for dish recognition. *IEEE transactions on multimedia* **17**(8), 1187–1199 (2015)