Self-Attention and Ingredient-Attention Based Model for Recipe Retrieval from Image Queries

Abstract

Direct computer vision based-nutrient content estimation is a demanding task, due to deformation and occlusions of ingredients, as well as high intra-class and low inter-class variability between meal classes. In order to tackle these issues, we propose a system for recipe retrieval from images. The recipe information can subsequently be used to estimate the nutrient content of the meal. In this study, we utilize the multi-modal Recipe1M dataset, which contains over 1 million recipes accompanied by over 13 million images. The proposed model can operate as a first step in an automatic pipeline for the estimation of nutrition content by supporting hints related to ingredient and instruction. Through self-attention, our model can directly process raw recipe text, making the upstream instruction sentence embedding process redundant and thus reducing training time, while providing desirable retrieval results. Furthermore, we propose the use of an ingredient attention mechanism, in order to gain insight into which instructions, parts of instructions or single instruction words are of importance for processing a single ingredient within a certain recipe. Attention-based recipe text encoding contributes to solving the issue of high intra-class/low inter-class variability by focusing on preparation steps specific to the meal. The experimental results demonstrate the potential of such a system for recipe retrieval from images. A comparison with respect to two baseline methods is also presented.

Introduction

Social media and designated online cooking platforms have made it possible for large populations to share food culture (diet, recipes) by providing a vast amount of food-related data. Despite the interest in food culture, global eating behavior still contributes heavily to diet-related diseases and deaths, according to the Lancet BIBREF0. Nutrition assessment is a demanding, time-consuming and expensive task.

Moreover, the conventional approaches for nutrition assessment are cumbersome and prone to errors. A tool that enables users to easily and accurately estimate the nutrition content of a meal, while at the same time minimize the need for tedious work is of great importance for a number of different population groups. Such a tool can be utilized for promoting a healthy lifestyle, as well as to support patients suffering food-related diseases such as diabetes. To this end, a number of computer vision approaches have been developed, in order to extract nutrient information from meal images by using machine learning. Typically, such systems detect the different food items in a picture BIBREF1, BIBREF2, BIBREF3, estimate their volumes BIBREF4, BIBREF5, BIBREF6 and calculate the nutrient content using a food composition database BIBREF7. In some cases however, inferring the nutrient content of a meal from an image can be really challenging - due to unseen ingredients (e.g. sugar, oil) or the structure of the meal (mixed food, soups, etc.).

Humans often use information from diverse sensory modalities (visual, auditory, haptic) to infer logical conclusions. This kind of multi-sensory integration helps us process complex tasks BIBREF8. In this study, we investigate the use of recipe information, in order to better estimate nutrient content of complex meal compositions. With the aim to develop a pipeline for holistic dietary assessment, we present and evaluate a method based on machine learning to retrieve recipe information from images, as a first step towards more accurate nutrient estimation. Such recipe information can then be utilized together with the volume of the food item to enhance an automatic system to estimate the nutrient content of complex meals, such as lasagna, crock pot or stew.

The performance of approaches based on machine learning relies heavily on the quantity and quality of the available data. To this end, a number of efforts have been made to compile informative datasets to be used for machine learning approaches. Most of the early released food databases were assembled only by image data for a special kind of meal. In particular, the first publicly available database was the Pittsburgh Fast-Food Image Dataset (PFID) BIBREF9, which contains only fast food images taken under

laboratory conditions. After the recent breakthrough in deep learning models, a number of larger databases were introduced. Bossard et al. BIBREF10 introduced the Food-101 dataset, which is composed of 101 food categories represented by 101'000 food images. This was followed by several image-based databases, such as the UEC-100 BIBREF11 and its augmented version, the UEC-256 BIBREF12 dataset, with 9060 food images referring to 100 Japanese food types and 31651 food images referring to 256 Japanese food types, respectively. Xu et al. BIBREF13 developed a specialized dataset by including geolocation and external information about restaurants to simplify the food recognition task. Wang et al. BIBREF14 introduced the UPMC Food-101 multi-modal dataset, that shares the same 101 food categories with the popular Food-101 dataset, but contains textual information in addition. A number of studies have been carried out utilizing the aforementioned databases, mainly for the task of food recognition. Salvador et al. BIBREF15 published Recipe1M, the largest publicly available multi-modal dataset, that consists of 1 million recipes together with the accompanying images.

The emergence of multi-modal databases has led to novel approaches for meal image analysis. The fusion of visual features learned from images by deep Convolution Neural Networks (CNN) and textual features lead to outstanding results in food recognition applications. An early approach for recipe retrieval was based on jointly learning to predict food category and its ingredients using deep CNN BIBREF16. In a following step, the predicted ingredients are matched against a large corpus of recipes. More recent approach is proposed by BIBREF15 and is based on jointly learning recipe-text and image representations in a shared latent space. Recurrent Neural Networks (RNN) and CNN are mainly used to map text and image into the shared space. To align the text and image embedding vectors between matching recipe-image pairs, cosine similarity loss with margin was applied. Carvalho et al. BIBREF17 proposed a similar multi-modal embedding method for aligning text and image representations in a shared latent space. In contrast to Salvador et al. BIBREF15, they formulated a joint objective function which incorporates the loss for the cross-modal retrieval task and a classification loss, instead of using the latent space for a multitask learning setup. To address the challenge of encoding long sequences (like

recipe instructions), BIBREF15 chose to represent single instructions as sentence embedding using the skip-thought technique BIBREF18. These encoded instruction sentences are referred to as skip-instructions and their embedding is not fine tuned when learning the image-text joint embedding.

In this study, we present a method for the joint learning of meal image and recipe embedding, using a multi-path structure that incorporates natural language processing paths, as well as image analysis paths. The main contribution of the proposed method is threefold: i) the direct encoding of the instructions, ingredients and images during training, making the need of skip instruction embedding redundant; ii) the utilization of multiple attention mechanisms (i.e. self-attention and ingredient-attention), and iii) a lightweight architecture.

Materials and Methods ::: Database

The proposed method is trained and evaluated on Recipe1M BIBREF15, the largest publicly available multi-modal food database. Recipe1M provides over 1 million recipes (ingredients and instructions), accompanied by one or more images per recipe, leading to 13 million images. The large corpus is supplemented with semantic information (1048 meal classes) for injecting an additional source of information in potential models. In the table in Figure FIGREF1, the structure of recipes belonging to different semantic classes is displayed. Using a slightly adjusted pre-processing than that in BIBREF15 (elimination of noisy instruction sentences), the training set, validation set and test set contain 254,238 and 54,565 and 54,885 matching pairs, respectively. In BIBREF15, the authors chose the overall amount of instructions per recipe as one criterion for a valid matching pair. But we simply removed instruction sentences that contain only punctuation and gained some extra data for training and validation.

Materials and Methods ::: Model Architecture

The proposed model architecture is based on a multi-path approach for each of the involved input data types namely, instructions, ingredients and images, similarly to BIBREF19. In Figure FIGREF4, the overall structure is presented. For the instruction encoder, we utilized a self-attention mechanism BIBREF20, which learns which words of the instructions are relevant with a certain ingredient. In order to encode the ingredients, a bidirectional RNN is used, since ingredients are an unordered list of words. All RNNs in the ingredients path were implemented with Long Short-Term Memory (LSTM) cells BIBREF21. We fixed the ingredient representation to have a length of 600, independent of the amount of ingredients. Lastly, the outputs of the self-attention-instruction encoder with ingredient attention and the output of the bidirectional LSTM ingredient-encoder are concatenated and mapped to the joint embedding space. The image analysis path is composed of a ResNet-50 model BIBREF22, pretrained on the ImageNet Dataset BIBREF23, with a custom top layer for mapping the image features to the joint embedding space. All word embeddings are pretrained with the word2vec algorithm BIBREF24 and fine tuned during the joint embedding learning phase. We chose 512-dimensional word embedding for our model with self-attention, whereas BIBREF19 and BIBREF17 chose a vector length of 300. In the following sections, more details about the aforementioned paths are presented.

Materials and Methods ::: Attention Mechanisms

The instruction encoder follows a transformer based encoder, as suggested by BIBREF20. Since we do not focus on syntactic rules, but mostly on weak sentence semantics or single words, we built a more shallow encoder containing only 2 stacked layers, where each of this layers contains two sub-layers. The first is the multi-head attention layer, and the second is a position-wise densely connected feed-forward network (FFN). Due to recipes composed of over 600 words as instructions, we decided to trim words per instruction sentence to restrict the overall words per recipe to 300. In order to avoid removing complete instructions at the end of the instruction table, we removed a fraction of words from each instruction, based on this instruction's length and the overall recipe-instruction length. This strategy reinforces the

neglect of syntactic structures in the instruction encoding process. With such a model, we can directly perform the instruction encoding during the learning process for the joint embedding, thus saving training time and reducing disk space consumption. The transformer-like encoder does not make use of any recurrent units, thus providing the opportunity for a more lightweight architecture. By using self-attention BIBREF20, the model learns to focus on instructions relevant to recipe-retrieval-relevant, parts of instructions or single instruction-words. Furthermore we gain insight into which instructions are important to distinguish recipes with similar ingredients but different preparation styles.

The instruction encoder transforms the sequence of plain word representations with added positional information to a sequence of similarity-based weighted sum of all word representations. The outputted sequence of the encoder exhibits the same amount of positions as the input to the instruction encoder (in our experiments 300). Each of this positions is represented by a 512-dimensional vector. To obtain a meaningful representation without a vast number of parameters, we reduced the number of word representations before the concatenation with the ingredient representation. For this reduction step, we implemented a recipe-embedding specific attention layer where the ingredient representation is used to construct n queries, where n is the amount of new instruction representation vectors. Each of these new representations is a composition of all previous word representations weighted by the ingredient attention score. Following, the ingredient attention process is formulated mathematically and is visually portrayed in Figure FIGREF4.

where $K(inst)$ and $V(inst)$ are linear mappings of the encoded instruction words, and $Q(ing)$ is a linear mapping of the ingredient representation and d_k is the dimensionality of linearly projected position vectors.

where b is the batch-size, p is the amount of word embeddings, w is the dimensionality of the wort embedding, h is the dimensionality of the space to where we project the word embeddings and queries,

q is the dimensionality of the ingredient representation and n is the amount of Ingredient Attention-based instruction representations. Ingredient Attention can be performed step-wise, similarly to the well known dimensionality reduction in convolution neural networks.

Materials and Methods ::: Loss function

To align text and image embeddings of matching recipe-image pairs alongside each other, we maximize the cosine distance between positive pairs and minimize it between negative pairs.

We have trained our model using cosine similarity loss with margin as in BIBREF19 and with the triplet loss proposed by BIBREF17. Both objective functions and the semantic regularization by BIBREF19 aim at maximizing intra-class correlation and minimizing inter-class correlation.

Let us define the text query embedding as $\phi ^q$ and the embedding of the image query as $\phi ^d$, then the cosine embedding loss can be defined as follows:

where $cos(x,y)$ is the normalized cosine similarity and $\alpha $ is a margin ($-1\leqslant \alpha \leqslant 1)$, that determines how similar negative pairs are allowed to be. Positive margins allow negative pairs to share at maximum $\alpha $ similarity, where a maximum margin of zero or negative margins allow no correlation between non matching embedding vectors or force the model to learn antiparallel representations, respectively. $\phi ^d$ is the corresponding image counterpart to $\phi ^q$ if $y=1$ or a randomly chosen sample $\phi ^d \in S \wedge \phi ^d \ne \phi ^{d(q)}$ if $y=-1$, where $\phi ^{d(q)}$ is the true match for $\phi ^q$ and S is the dataset we sample from it. Furthermore, we complement the cosine similarity with cross-entropy classification loss (L_{reg}), leading to the applied objective function.

with c_r and c_v as semantic recipe-class and semantic image-class, respectively, while $c_r=c_v$ if the food image and recipe text are a positive pair.

For the triplet loss, we define $\phi ^q$ as query embedding, $\phi ^{d+}$ as matching image counterpart and $\phi ^{d-}$ as another random sample taken from S. Further $\phi ^{d_{sem}+} \in S \wedge \phi ^{d_{sem}+} \ne \phi ^{d(q)}$ is a sample from S sharing the same semantic class as $\phi ^q$ and $\phi ^{d_{sem}-}$ is a sample from any other class. The triplet loss is formulated as follows:

where $\beta \in [0,1]$ weights between quadratic and linear loss, $\alpha \in [0,2]$ is the margin and $\gamma \in [0,1]$ weights between semantic- and sample-loss. The triplet loss encourages the embedding vectors of a matching pair to be larger by a margin above its non-matching counterpart. Further, the semantic loss encourages the model to form clusters of dishes, sharing the same class. We chose $\beta $ to be 0.1, $\alpha $ to be 0.3 and $\gamma $ to be 0.3.

Materials and Methods ::: Training configuration

We used Adam BIBREF25 optimizer with an initial learning rate of 10^{-4}. At the beginning of the training session, we freeze the pretrained ResNet-50 weights and optimize only the text-processing branch until we do no longer make progress. Then, we alternate train image and text branch until we switched modality for 10 times. Lastly, we fine-tune the overall model by releasing all trainable parameters in the model. Our optimization strategy differs from BIBREF19 in that we use an aggressive learning rate decay, namely exponential decay, so that the learning rate is halved all 20 epochs. Since the timing of freezing layers proved not to be of importance unless the recipe path is trained first, we used the same strategy under the cosine distance objective BIBREF19 and for the triplet loss BIBREF17.

Experimental Setup and Results

Recipe1M is already distributed in three parts, the training, validation and testing sets. We did not make any changes to these partitions. Except with our more sensitive preprocessing algorithm, we accept more recipes from the raw corpus. BIBREF19 used 238,399 samples for their effective training set and for the validation and testing set 51,119 and 51,303 samples, respectively. By filtering out noisy instructions sentences (e.g. instructions containing only punctuation) we increased the effective dataset size to 254,238 samples for the training set and 54,565 and 54,885 for the validation and testing sets, respectively.

Similarly to BIBREF19 and BIBREF17, we evaluated our model on 10 subsets of 1000 samples each. One sample of these subsets is composed of text embedding and image embedding in the shared latent space. Since our interest lies in the recipe retrieval task, we optimized and evaluated our model by using each image embedding in the subsets as query against all text embeddings. By ranking the query and the candidate embeddings according to their cosine distance, we estimate the median rank. The model's performance is best, if the matching text embedding is found at the first rank. Further, we estimate the recall percentage at the top K percent over all queries. The recall percentage describes the quantity of queries ranked amid the top K closest results. In Table TABREF11 the results are presented, in comparison to baseline methods.

Both BIBREF19 and BIBREF17 use time-consuming instruction text preprocessing over the skip-thought technique BIBREF18. This process doubles the overall training time from three days to six days using two Nvidia Titan X GPU's. By using online-instruction encoding with the self-attention encoder, we were able train the model for its main task in under 30 hours. Furthermore, the proposed approach offers more flexibility for dataset alterations.

Qualitative results such as recipe retrieval, quality of the cluster formation in the joint embedding space and heat maps of instruction words are more important than the previously mentioned benchmarking

scores. Depending on meal type, all baseline implementations as well as our Ingredient Attention based model exhibit a broad range of retrieval accuracy. In Figure FIGREF16 we present a few typical results on the intended recipe retrieval task.

AdaMine BIBREF17 creates more distinct class clusters than in BIBREF19. In Figure FIGREF12, we demonstrate the difference in cluster formation using the aforementioned Methods for our Ingredient Attention. We visualize the top ten most common recipe classes in Recipe1M using t-SNE BIBREF26. Since chocolate chip, peanut butter, cream cheese and/or ice cream are used as ingredients in desserts, due to semantic regularization inside the triplet loss, clusters of sweet meals are close together (Figure FIGREF12 top right corner).

We use heat maps on instruction words as tool to visualize words relevant to ingredient-lists in plain instruction text. In Figure FIGREF15, we demonstrate how easily we can achieve insight into the models decision making.

Conclusions

In this paper, we have introduced self-attention for instruction encoding in the context of the recipe retrieval task and ingredient attention for disclosing ingredient dependent meal preparation steps. Our main contribution is the aforementioned ingredient attention, empowering our model to solve the recipe retrieval without any upstream skip instruction embedding, as well as the light-weight architecture provided by the transformer-like instruction encoder. On the recipe retrieval task, our method performs similarly to our baseline implementation of BIBREF17. Regarding training time on the other hand, we increased the efficiency significantly for cross-modal based retrieval methods. There is no need for a maximum number of instructions for a recipe to be considered as valid for training or testing; only for total words, making more samples of the large Recipe1M corpus usable for training. Through ingredient

attention, we are able to unveil internal focus in the text processing path by observing attention weights. Incorporation of new samples in the train set can be done by retraining just one model. Overall, an accurate and flexible method for recipe retrieval from meal images could provide downstream models (e.g. automatic nutrient content estimation) with decisive information and significantly improve their results.