Multi-modal Multi-task Learning for Automatic Dietary Assessment

Qi Liu[1], Yue Zhang[1], Zhenguang Liu[4], Ye Yuan[1], Li Cheng[2,3], Roger Zimmermann[3]
1. Singapore University of Technology and Design
2. Bioinformatics Institute, A*STAR, Singapore
3. School of Computing, National University of Singapore
4. Zhejiang Gongshang University
{qi_liu, yue_zhang, ye_yuan}@sutd.edu.sg, liuzhenguang2008@gmail.com,
chengli@bii.a-star.edu.sg, rogerz@comp.nus.edu.sg

Abstract

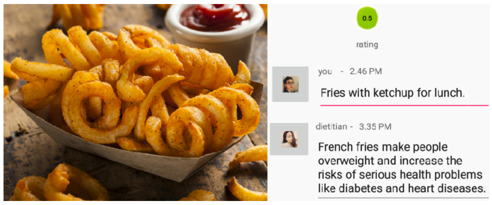
We investigate the task of automatic dietary assessment: given meal images and descriptions uploaded by real users, our task is to automatically rate the meals and deliver advisory comments for improving users' diets. To address this practical yet challenging problem, which is multi-modal and multi-task in nature, an end-to-end neural model is proposed. In particular, comprehensive meal representations are obtained from images, descriptions and user information. We further introduce a novel memory network architecture to store meal representations and reason over the meal representations to support predictions. Results on a real-world dataset show that our method outperforms two strong image captioning baselines significantly.

Introduction

Chronic diseases such as cardiovascular diseases, type 2 diabetes, metabolic syndromes and cancers are the leading killers in developed countries and have been increasingly rampant in developing nations (Alwan 2011). Nowadays, obesity, diabetes and hypertension are even common among children. Improving diets is a solution to this epidemics of metabolic diseases that are inundating the world (Roberts and Barnard 2005). Any successful strategy to improve a person's diet and lifestyle is expected to pay off with improved health in the long term.

Since mobile devices are ubiquitous today and deeply impact daily life, dietary interventions conducted by mobile applications hold the promise for long-term diet management (Rebedew 2014; Goyal et al. 2017). An example of these applications is shown in Figure 1. Here *users* can upload an image and append a short description for reporting the meal they are consuming, and *dietitians* can respond to the users with a meal rating (a single quantitative score for rating the meal from very unhealthy to very healthy) and detailed comments. Compared to traditional dietary advisory methods, mobile applications are more accessible and well-suited for dietary intervention.

However, relying on manual assessments by dietitians is both expensive and time-consuming. To this end, tools that can assist dietitians or even automate the advisory process can be highly useful. In this paper, we introduce a new problem, utilizing algorithms to automatically assess uploaded images and descriptions for diet management. The problem is challenging in several aspects:



you - 2.46 PM

Fries with ketchup for lunch.

dietitian - 3.35 PM

French fries make people overweight and increase the risks of serious health problems like diabetes and heart diseases.

Figure 1: Application example. Dietitians respond to the users with meal ratings and comments.

- The algorithm has to handle multiple modalities. In addition to their images and text descriptions, meals should also be evaluated based on user information (e.g. less salt should be taken by users with hypertension), user characteristics such as gender, age, blood sugar level and smoking or not must be considered.
- The problem is multi-task innate in that the algorithm needs to provide both meal ratings and advisory comments, which should be instructive for diet management.
- User-uploaded data are highly diverse. Diversities in food cultures and cooking styles bring in challenges for image evaluation. In addition, the descriptions can be written with poor grammatical quality, and user information mainly consists of categorical variables, which are highly sparse in practice.

Motivated by the success of neural models for feature representations (Krizhevsky, Sutskever, and Hinton 2012; Hochreiter and Schmidhuber 1997; Karpathy and Fei-Fei 2015; He and Chua 2017), we devise a multi-modal multi-task learning framework to solve the problem. In particular, we utilize a convolutional neural network (Simonyan and Zisserman 2014), a bidirectional long short term memory (Schmidhuber 2005) and a neural factorization machine (He and Chua 2017) to extract features from an image, a description and their corresponding user information, respectively. The three feature representations are concatenated to obtain a comprehensive *meal representation* for assessment.

In addition, since historical meal information can provide valuable contextual clues for predictions (e.g. what kind of food that the users usually consume, such as western food and Asian food), we devise a new memory network architecture (Weston, Chopra, and Bordes 2014) to hold historical meal information. A *background meal representation* is obtained by attending over memory networks. The meal representation enhanced by the background meal representation are fed into prediction layers. Finally, we apply a long short term memory decoder for generating comments.

Experiments on a real-world dataset show that our model outperforms two strong image captioning baseline models significantly. Furthermore, case studies indicate that our algorithm can reliably generate meal ratings and comments.

Our contributions are twofold:

- To our knowledge, we are the first to use both meal ratings and comments for dietary assessment.
- We propose a novel end-to-end framework for automatic dietary assessment, utilizing multiple modalities as inputs. A novel memory network architecture is devised, which holds users' historical meal information and provides supporting evidences for online predictions.

Related Work

Dietary Assessment: There is a stream of work utilizing images for dietary assessment. Dehais et al. (2015) utilize dish detection and image segmentation techniques to recognize food categories and portion sizes from images. Liang et al. (2017) utilize R-CNN to estimate calories from images. Hassannejad et al. (2017) design a pipeline system, which consults a nutrient database to generate comments after food and portion size recognitions, employing some predefined templates. Our work is different from existing work in three aspects. (1) We propose using both meal ratings and advisory comments for dietary assessment, instead of solely recognizing food categories, portion sizes or calories. The advisory comments are generated with a long short term memory, thus being more customized and flexible compared to template-based methods. (2) Our work is an end-to-end framework, reducing error propagations in pipeline systems. (3) We utilize multiple modalities (images, descriptions and user information) for predictions.

Our work is also related to previous work on multi-modal learning, multi-task learning and memory networks.

Multi-modal Learning: Predicting image ratings and comments using multiple modalities are related to multi-modal fusion and translation, respectively. Multi-modal fusion aims to join information from two or more modalities to perform predictions (classification or regression). For example, Chen and Jin (2015) explore methods utilizing bi-LSTM to fuse audio and visual signals for predicting values of the emotion dimensions, arousal and valence. Yang et al. (2016) fuse visual, audio and textual features for video classifications.

Multi-modal translation aims to translate one or more input modalities to another modality. Barbu et al. (2012) propose a pipeline framework to generate natural sentences for describing videos. Mansimov et al. (2015) introduce an attentional model to generates images from natural language descriptions. Relatedly, image captioning (Karpathy and Fei-Fei 2015; Xu et al. 2015) generates descriptions from images.

Different from the above work, we build a multi-task learning framework, which jointly predicts meal ratings and comments from images, descriptions and user information.

Multi-task Learning jointly learns multiple tasks to improve the generalization performance of all tasks (Zhang and Yang 2017). Here we focus on neural models. One stream of work assumes that each task has its own set of parameters, and the parameters of the tasks are regularized to capture task relations. Duong (2015) use l_2 norm for regularization, while Yang et al. (2016) introduces trace norm. Another stream of work achieves multi-task learning by parameter sharing. Nam and Han (2016) propose using convolutional neural networks as shared layers for visual tracking. Liu, Qiu and Huang (2016) utilize LSTM as shared layers for text classification. Our work belongs to the second stream of work, where two tasks (meal ratings and comments) share meal representations.

Memory Networks reason with inference components combined with a long-term memory component. Weston et al. (2014) devise a memory network to explicitly store the entire input sequences for question answering. An end-to-end memory network is further proposed by Sukhbaatar et al. (2015) by storing embeddings of input sequences, which requires much less supervision compared to Weston et al. (2014). Kumar et al. (2016) introduces a general dynamic memory network, which iteratively attends over episodic memories to generate answers. Xiong et al. (2016) extends Kumar et al. (2016) by introducing a new architecture to cater image inputs and better capture input dependencies. In similar spirits, our memory network stores meal representations for obtaining background meal representation by attention.

Problem Definition

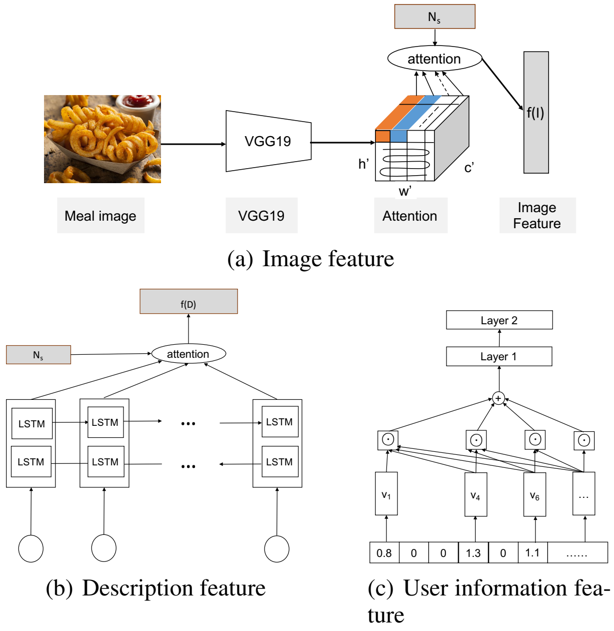
Formally, the input scenario contains a user S, with user information P, who uploads an image I and a description D. The image I is of shape $h \times w \times c$ (h, w and c are image height, weight and channel, respectively). The description D is a sequence of words $d_1, d_2 ... d_{|D|}$, each d_i being drawn from a vocabulary V. The user information P is a vector of size n. Our task is to automatically assess the inputs, obtaining a numerical meal rating G and a detailed advisory comment E, which is a sequence of words $e_1, e_2 ... e_{|E|}$, and e_1 and $e_{|E|}$ are a special START and a special END token, respectively.

Feature Extraction

Figure 2 shows the feature extraction models for images, descriptions and user information.

Image Features

Our model for image feature extraction is shown in Figure 2 (a). Given an image $I \in R^{h \times w \times c}$ uploaded by a user S, we use VGG-19 (Simonyan and Zisserman 2014) to extract its



(a) Image feature

(b) Description feature

(c) User information feature

Figure 2: Feature extraction models

image feature. The basic idea of VGG-19 is to use a series of convolution and pooling layers followed by fully connected layers. We take the output I' from the last pooling layer, which is of shape, $h' \times w' \times c'$. As a result, I' divides the image into $h' \times w'$ local regions, each local region I'_i being represented as a c' dimensional feature vector.

Not all local regions contribute equally for predictions. The local regions corresponding to food objects should be paid more attention to. Neural attention (Bahdanau and Bengio 2014) has been shown useful for tasks such as machine translation (Bahdanau and Bengio 2014), reading comprehension (Cheng, Dong, and Lapata 2016) and image captioning (Xu et al. 2015). Formally, the inputs to the attention mechanism are a query and a set of key-value pairs, where the query, keys, values are all vectors. The output is calculated as a weighted sum of the value vectors, where the weights are obtained by calculating the similarities (e.g. cosine similarities) between the query and the keys. In this paper, keys are equal to their corresponding values.

We leverage *user descriptors* as queries for calculating the weights of the local regions, which are introduced to explicitly capture user characteristics, parametrized by a matrix $N \in R^{K \times m}$. Suppose that there are m users in the training set, the user descriptor of each user corresponds to one column of N, denoted as $N_S \in R^K$ for a user S. The user descriptors are automatically learned during training. During testing, for users not appearing in the training set, we use the average of the m user descriptors in N as his/her user descriptors.

We use the user descriptor N_S (the query) to attend over the $h' \times w'$ local regions I'_i (the keys/values) for obtaining a representation of I. Since I'_i and N_S may not be of the same lengths, we apply additive attention (Bahdanau and Bengio 2014), which uses a feed-forward network with a single hidden layer. Formally, The image feature $f(I)$ is calculated as:

$$f(I) = \sum_{i=1}^{h' \times w'} a_i I'_i \quad s.t. \sum_{i=1}^{h' \times w'} a_i = 1 \tag{1}$$

The weight a_i reflects the importance of I'_i with respect to N_S and is evaluated as:

$$\begin{aligned} l_i &= v^T tanh(AN_S + QI'_i) \\ a_i &= \frac{exp(l_i)}{\sum_{j=1}^{h' \times w'} exp(l_j)} \end{aligned} \tag{2}$$

Here $A \in R^{2K \times K}$, $Q \in R^{2K \times c'}$ and $v \in R^{2K}$ are parameters of additive attention. A and Q linearly project N_i and h_t to a hidden layer of length $2K$, respectively. The projected space is set as $2K$ empirically, since we find it beneficial to project the vectors into a larger layer. v serves as the output layer. Softmax is applied to normalize l_i. We use $f(I) \in R^{c'}$ as the image feature.

Textual Features

Our model for textual features is shown in Figure 2 (b). Given a description D upload by a user S, its word sequence $d_1, d_2 ... d_{|D|}$ is fed into a word embedding layer to obtain embedding vectors $x_1, x_2 ... x_{|D|}$. The word embedding layer is parameterized by an embedding matrix $E_w \in R^{K \times |V|}$, where K is the embedding dimension, and $|V|$ is the vocabulary size.

To acquire a semantic representation of D, Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber 1997) is utilized to transform embedding vectors $x_1, x_2 ... x_{|D|}$ to a sequence of hidden states $h_1, h_2 ... h_{|D|}$. We use the variation of Hochreiter and Schmidhuber (1997), which takes advantages of an input gate, a forget gate and an output gate, denoted as i_t, f_t and o_t, respectively, to control information flow. A LSTM cell incrementally consumes one input x_t at each time step t. Given an x_t, the previous hidden state h_{t-1} and cell state c_{t-1}, the LSTM cell computes the next hidden state h_t and the next cell state c_t as:

$$\begin{aligned} i_t &= \sigma(W^{(i)} x_t + U^{(i)} h_{t-1} + b^{(i)}) \\ f_t &= \sigma(W^{(f)} x_t + U^{(f)} h_{t-1} + b^{(f)}) \\ o_t &= \sigma(W^{(o)} x_t + U^{(o)} h_{t-1} + b^{(o)}) \\ u_t &= tanh(W^{(u)} x_t + U^{(u)} h_{t-1} + b^{(u)}) \\ c_t &= i_t \odot u_t + f_t \odot c_{t-1} \\ h_t &= o_t \odot tanh(c_t) \end{aligned}$$

Here, σ denotes the sigmoid function and $\odot$ is the element-wise multiplication.

A bidirectional extension (Schmidhuber 2005) is applied to capture sentence-level semantics both left-to-right and right-to-left. As a result, two sequences of hidden states are obtained, denoted as $\overrightarrow{h}_1, \overrightarrow{h}_2 ... \overrightarrow{h}_{|D|}$ and $\overleftarrow{h}_1, \overleftarrow{h}_2 ... \overleftarrow{h}_{|D|}$, respectively. We concatenate $\overrightarrow{h}_t$ and $\overleftarrow{h}_t$ at each time step to obtain the final hidden states $h_1, h_2 ... h_{|D|}$, which are of sizes $2K$.

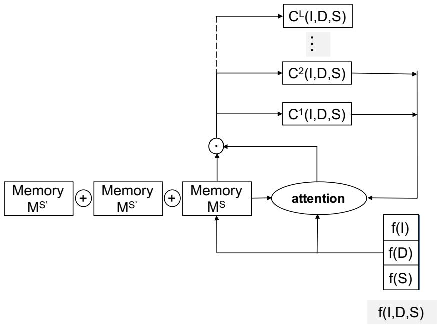


Figure 3: Memory network for reasoning

To select the most salient words from a description, we use N_S as the query to attend over hidden states $h_1, h_2 ... h_{|D|}$, using Equation 1 and 2 to obtain a final textual feature vector $f(D) \in R^{2K}$.

User Information Features

Since the meals are evaluated based on user information, a user information feature $f(S)$ is obtained for S. Our model for user information feature extraction is shown in Figure 2 (c).

User information consists of a variety of categorical variables (e.g. gender, occupation, disease, smoking or not). A common practice is to convert these categorical variables to a set of binary features via one-hot encoding (Shan, Hoens, and Jiao 2016). However, the resultant first-order user information feature $P \in R^n$ can be highly sparse. To remedy the problem, it is essential to take advantage of the interactions between features. To this end, factorization machines (Rendle 2010) can be a promising family of algorithms, which learns feature interactions from raw data automatically. Neural factorization machine improves over naive factorization machines and has been proven to be effective in capturing high-order feature interactions (He and Chua 2017). Given P, it utilizes a bi-pooling layer parametrized by a feature embedding matrix $B \in R^{K \times n}$ to capture second-order interactions:

$$\textit{bi-pooling}(P) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} P_i B_i \odot P_j B_j \tag{3}$$

Here P_i and P_j are the ith and jth features of P, respectively. The ith column of B is the feature embedding B_i of P_i. Thus, B_i and B_j are feature embeddings corresponding to P_i and P_j, respectively. $\textit{bi-pooling}(P) \in R^K$ is the second-order feature representation.

To capture higher-order interactions, we further apply two fully connected hidden layers with $tanh$ activation function on $\textit{bi-pooling}(P)$, obtaining a high-order user information feature $f(S) \in R^K$.

Reasoning with Memory Network

We concatenate $f(I)$, $f(D)$ and $f(S)$ to obtain a meal representation, $f(I, D, S) \in R^{(c'+3K)}$. Further, we assume that users' historical meal information is highly instructive, since valuable background evidences such as eating habits and written styles of descriptions are revealed. Thus, reasoning capabilities on historical meal information can be helpful. Memory networks (Weston, Chopra, and Bordes 2014; Sukhbaatar et al. 2015) have been shown effective in storing evidences for question answering. In this paper, we devise a novel memory network architecture shown in Figure 3, which stores historical meal representations to support online predictions.

Formally, a memory network $M^S \in R^{(c'+3K) \times |S|}$ is introduced for each user S, where $|S|$ denotes the number of meals uploaded by S.

Memory Network: Suppose that $f(I, D, S)$ is the ith meal uploaded by S, we directly set $f(I, D, S)$ as the ith column of M^S. Formally,

$$M_i^S = f(I, D, S) \tag{4}$$

As a result, the columns of M^S are historical meal representations, ordered by their upload time.

Dealing with Data Scarcity: Data scarcity is common among inactive users. We propose using the memory networks of the most similar users to enhance the memory network M^S with few columns. The similarity of two memory networks, M^S and $M^{S'}$ is calculated as:

$$\begin{aligned} sim(M^S, M^{S'}) &= \frac{\sum_i^{|S|} \sum_j^{|S'|} M_i^S \cdot M_j^{S'}}{|S||S'|} \\ &= \frac{\sum_i^{|S|} M_i^S}{|S|} \cdot \frac{\sum_j^{|S'|} M_j^{S'}}{|S'|} \end{aligned} \tag{5}$$

where $\cdot$ denotes dot product. The similarity value is the average dot product between the columns of M^S and $M^{S'}$. We utilize the r memory networks with the largest similarities and concatenate the r memory networks with M^S to enhance it. We denote the concatenation result as M_+^S.

Since calculating the similarities between each pair of memory networks is expensive, we use locality sensitive hashing (LSH) (Shrivastava and Li 2014) to approximately obtain the similarities. As $sim(M^S, M^{S'})$ is the dot product between $\frac{\sum_i^{|S|} M_i^S}{|S|}$ and $\frac{\sum_j^{|S'|} M_j^{S'}}{|S'|}$ in Equation 5, for each $M^{S'}$, its representation $\frac{\sum_j^{|S'|} M_j^{S'}}{|S'|}$ is indexed to LSH tables. For a query $\frac{\sum_i^{|S|} M_i^S}{|S|}$, the r most similar memory networks measured by dot product are obtained in sub-linear time complexity (Shrivastava and Li 2014), which greatly improves efficiency in practice.

When a memory network is large, we simply use the most recently uploaded meals (meals uploaded in the last three months are preserved, empirically) as historical information.

Obtaining a Background Meal Representation: Given a meal representation $f(I, D, S)$, we obtain a background mean representation $C(I, D, S)$ to support the prediction by attending over its enhanced memory M_{+}^{S}:

$$
\begin{aligned}
C^{1}(I,D,S) &= M_{+}^{S} \textit{softmax}((M_{+}^{S})^{T} f(I,D,S)) \\
C^{2}(I,D,S) &= M_{+}^{S} \textit{softmax}((M_{+}^{S})^{T} C^{1}(I,D,S)) \\
&\ldots\ldots \\
C^{L}(I,D,S) &= M_{+}^{S} \textit{softmax}((M_{+}^{S})^{T} C^{L-1}(I,D,S))
\end{aligned}
\tag{6}
$$

Dot product attention (Vaswani and Shazeer 2017) is applied here, which is faster and more space-efficient compared to additive attention, since it can be implemented using highly optimized matrix multiplication. Dot products are performed between $f(I,D,S)$ and each column of M_{+}^{S} and the scores are normalized using the softmax function. $C^{1}(I,D,S)$ is a weighted sum of M_{+}^{S}'s columns. The process repeats until the Lth-step reasoning state $C^{L}(I,D,S)$ is obtained. We use multiple step reasoning in that it has been proven that the memory network may need to be consulted several times to obtain contextual information (Xiong, Merity, and Socher 2016).

In summary, each meal representation $f(I,D,S)$ is stored into M^{S} using Equation 4. M^{S} is enhanced to M_{+}^{S} by finding its most similar memory networks using LSH. Finally, $C^{L}(I,D,S)$ is obtained by attending over M_{+}^{S}.

Prediction

We utilize element-wise addition of the meal representation and its background meal representation, $f(I,D,S) + C^{L}(I,D,S)$ for predictions, denoted as $f_{+}(I,D,S) \in R^{(c'+3K)}$. Meal rating and comment are jointly learned, since the two tasks can benefit from multi-task learning. The reasons are twofold: (1) Multi-task learning acts as a regularizer, which reduces the risk of overfitting (Nam and Han 2016). (2) The two tasks are closely related and data are augmented implicitly by joint learning (Wang and Zhang 2017).

Meal Rating

We introduce a prediction layer for estimating meal ratings, parametrized by a vector $z \in R^{(c'+3K)}$ and a bias $b_{mr} \in R$. Formally, meal rating estimations are calculated as:

$$\hat{G} = z^{T} f_{+}(I,D,S) + b_{mr} \tag{7}$$

The mean square error is used as the loss function:

$$loss_{mr} = (G - \hat{G})^{2} \tag{8}$$

Meal Comment

Motivated by the success of LSTM in machine translation (Bahdanau and Bengio 2014), image captioning (Karpathy and Fei-Fei 2015) and abstractive summarization (Wang and Zhang 2017), we utilize LSTM to generate comments. During training, we condition the generative process on $f_{+}(I,D,S)$ (i.e. using $f_{+}(I,D,S)$ as LSTM's initial cell state c_{0}). In addition, we sequentially input comment E's word sequence $e_{1}, e_{2} \ldots e_{|E|-1}$ to LSTM, obtaining hidden states $h_{1}, h_{2} \ldots h_{|E|-1}$. For each h_{i}, we generate a probability distribution over the vocabulary V:

$$p_{i} = softmax(W_{mc} h_{i} + b_{mc}) \tag{9}$$

where $h_{i} \in R^{(c'+3K)}$, and $W_{mc} \in R^{|V| \times (c'+3K)}$ and $b \in R^{|V|}$. The probability of word e_{j} in p_{i}, denoted as $p_{i}(e_{j})$, represents the probability of generating e_{j} in the ith step. The loss function maximizes the log probabilities of $e_{2} \ldots e_{|E|}$ in $p_{1} \ldots p_{|E|-1}$, respectively:

$$loss_{mc} = \sum_{i=1}^{|E|-1} logp_{i}(e_{i+1}) \tag{10}$$

# user	# record	description	rating	comment
283	42733	9.3 ± 3.2	1.7 ± 0.29	27.6 ± 4.2

Table 1: Dataset statistics

During testing, to generate a comment, we condition on $f_{+}(I,D,S)$ and use the special START token as the first input. Next, we pick the word with the maximum probability as the next word and use it as the second input. The process repeats until the END token is picked.

Joint Learning

To jointly learn the meal rating and meal comment tasks, we minimize the following joint loss function, which linearly interpolates $loss_{mr}$ and $loss_{mc}$, controlled by λ:

$$loss = loss_{mr} - \lambda loss_{mc} \tag{11}$$

Experiments

Dataset and Preprocessing

Our dataset is obtained from a mobile application for diet management, which allows users to create their accounts with personal information and report their diets by taking photos and attaching text descriptions. A dietitian team evaluates the uploaded photos and descriptions by rating the meals between 0 (very unhealthy) and 3 (very healthy) and attaching their detailed comments about the ratings. The dataset statistics are shown in Table 1, which ranges from 06/07/2016 to 09/01/2017. There are 283 anonymous users for privacy. The users uploaded $42,733$ meals, which are evaluated by dietitians. The attached descriptions have an average length of 9.3 words, and its standard deviation is 3.2. The averages of dietitian ratings and comments are 1.7 points and 27.6 words, respectively.

We resize the images to $448 \times 448 \times 3$. For the descriptions, since the texts are noisy, we use NLTK[1] to perform spelling error correction and text normalization. For user information, we fill out missing values by setting them as average values (numerical variables) or values with maximum frequencies (categorical variables). Next, we use one-hot encoding to transform the variables into first-order user information vectors, which are of sizes 1574. We select 70%, 10% and 20% of the meals as training, development and testing sets, respectively, according to their upload time, since our algorithm intends to evaluate future meals according to users' historical activities.

[1] http://www.nltk.org/

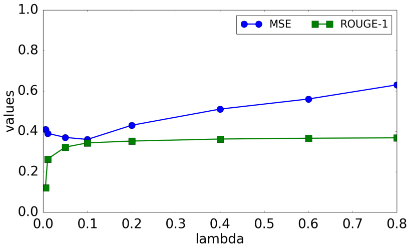


Figure 4: Effect of lambda on MAE and ROUGE-1

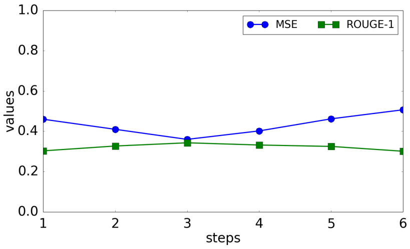


Figure 5: Effect of L on MAE and ROUGE-1

Method	*MAE*	*ROUGE-1*
BRNN	-	0.247
BRNN+mr	0.473	0.258
BRNN+all	0.458	0.278
SAT	-	0.263
SAT+mr	0.462	0.274
SAT+all	0.442	0.285
MMDA-att-mem	0.452	0.265
MMDA-mem	0.424	0.285
MMDA-mc	0.403	-
MMDA-mr	-	0.317
MMDA	**0.387***	**0.335***

Table 2: Meal rating and comment prediction. * denotes statistical significance using t-test ($p < 0.01$), compared to the second best.

Model Settings

For the images, we take out the outputs from the last pooling layer of VGG-19, which are of sizes $14 \times 14 \times 512$. Thus, the outputs divide each image into 14×14 local regions, which are of lengths 512. For the descriptions, the vocabulary size is set as $10,000$. K is empirically set as 200. We extend these memory networks that have fewer than 40 columns with its 3 most similar memory networks.

We use stochastic gradient descent with mini batch sizes of 50. Dropout (Krizhevsky, Sutskever, and Hinton 2012) is used to avoid overfitting, and the dropout rate is set as 0.5. We use AdaGrad as the optimizer, and the initial learning rate for AdaGrad is set as 0.5. Also, the gradient clipping (Karpathy and Fei-Fei 2015) is adopted to prevent gradient exploding and vanishing, where gradients larger than 5 are rescaled. All experiments are conducted on a PC with a Intel 3.4 GHz CPU, a 4 GB memory and a 8 GB 1080 GPU.

We use the mean absolute error (the lower, the better), $|G - \hat{G}|$, denoted as MAE to evaluate the goodness of meal ratings. ROUGE-1[2] (Lin 2004) (the higher, the better) is used to evaluate meal comments. Here Rouge-n measures the n-gram recall between a algorithm-generated meal comment $\hat{E}$ and a dietitian comment E.

Sensitivity Test

We study how to set the loss controller λ in Equation 11 and the number of reasoning steps L in Equation 6.

Study of λ: We set the reasoning step $L = 3$ and tune λ in $[0.005, 0.01, 0.05, 0.1, 0.2, 0.4, 0.8]$, recording MAE and ROUGE-1 on the development set. The results are shown in Figure 4.

When λ is small, the algorithm resembles single-task learning (i.e. optimizing meal ratings). With the increase of λ, MAE of meal ratings become smaller, which sheds light on the effectiveness of multi-task learning, while peaking at 0.1. In addition, ROUGE-1 keeps going up, but the slopes decrease. Thus, we set $\lambda = 0.1$ in the experiments.

Study of L: We change reasoning steps from 1 to 6 and record MAE and ROUGE-1 on the development set. The results are shown in Figure 5.

We observe that when $L = 3$, both MAE an ROUGE-1 achieve the best performances. It indicates that more hops can be useful for capturing more abstract contextual information to improve performances. However, when $L > 3$, the model becomes over-fitted, leading to worse performances. As a result, we use $L = 3$ in the experiments.

Quantitative Results

For comparison, we use two image captioning methods (Karpathy and Fei-Fei 2015; Xu et al. 2015) as baselines.

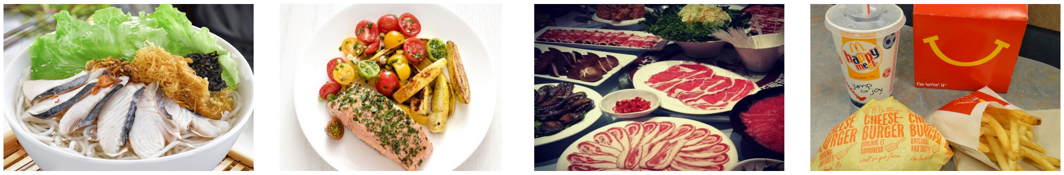
BRNN is our implementation of Karpathy et al. (2015), which utilizes a region convolutional neural network to extract image features of sizes 4096 and a recurrent neural network (RNN) for generating comments.

SAT is our implementation of Xu et al. (2015), which extends **BRNN** using a attention model. **SAT** utilizes a convolutional neural network to obtain 14×14 local regions, each of lengths 512, and attends over these local regions when generating comments with LSTM.

BRNN+mr and **SAT+mr** are our extensions to generate meal ratings. For **BRNN**, we simply input the 4096 feature vectors to Equation 7 for generating ratings. For **BRNN**, the initial memory state and hidden state of LSTM generated from the average of 14×14 local regions are fed into Equation 7.

Since the above baselines only utilize image features for prediction, we take out the pre-trained description and user information features, $f(D)$ and $f(S)$ from our algorithm and employ these features to enhance **BRNN+mr** and **SAT+mr**. We enhance the image features by concatenating

[2] `https://github.com/andersjo/pyrouge/tree/master/tools/ROUGE-1.5.5`



(a) Mee Hoon with fish comment: Proportion between mee hoon and vegetables is nice and the fish portion contains omega 3, that is good for your heart! rating: 2.8

(b) Salmon with veg for lunch comment: A nice combine of salmon and vegetable. It would be better to have a small portion of brown rice. rating: 2.6

(c) Having hotpot buffet with three friends comment: Thank you for your meal log! The fat is high and portion is large. Try to eat less. rating: 0.4

(d) McDonald's lunch comment: McDonald's is not a good choice! Lack of fibre. The fat and salt are bad for heart. Try some salads! rating: 0.5

Figure 6: Case study

it with $f(D)$ and $f(S)$ to extend **BRNN+mr**. For **SAT+mr**, at each attention step, we concatenate the image features with $f(D)$ and $f(S)$. In addition, we append the average of 14×14 local regions with $f(D)$ and $f(S)$, when generating the initial memory state and hidden state of LSTM. We denote these two algorithms as **BRNN+all** and **SAT+all**, respectively.

We also compare the results with several variants of our algorithm, denoted as **MMDA**:

MMDA-att-mem removes attention and memory network modules for predictions. The 14×14 local regions and LSTM hidden states $h_1, h_2 ... h_{|D|}$ are averaged to generate image and description features, respectively. The features are concatenated and fed into Equation 7 and 9 for generating ratings and comments directly.

MMDA-mem removes memory network module. Only meal representations are fed into the prediction layers.

MMDA-mc is an multi-modal single-task baseline, which removes the layers for meal comment predictions.

MMDA-mr is an multi-modal single-task baseline, which removes the layers for meal rating predictions.

The results on the test set are shown in Figure 2. **BRNN+mr** and **SAT+mr** outperforms **BRNN** and **SAT**, respectively, which confirms the effectiveness of multi-task learning on reducing over-fitting and providing additional information. **BRNN+all** and **SAT+all** further outperforms **BRNN+mr** and **SAT+mr**, respectively, which shows that multi-modal learning can improve diet assessment. In addition, **SAT**, **SAT+mr** and **SAT+all** can achieve better performance compared to **BRNN**, **BRNN+mr** and **BRNN+all**, respectively, which reveals the effectiveness of attentional model in extracting features.

For our algorithms, **MMDA-att-mem** performs worst. **MMDA-mem** improves over **MMDA-att-mem** by attention over images and texts. **MMDA** outperforms **MMDA-mem** by a large margin, which confirms that historical meal information is instructive, and the memory network can effectively provide background information. In addition, **MMDA** outperforms **MMDA-mc** and **MMDA-mr**, which further confirms the efficacy of multi-task learning.

Case Study

We perform a case study and randomly select some examples shown in Figure 6 (user information is not displayed for privacy). We observe that our algorithm can generate reasonable ratings and comments, which are instructive for diet management. Another observation is that the comments and ratings are predicted, considering user information (e.g. in Figure 6 (d), the effects on the heart are mentioned since the user has a history of heart-related diseases).

We also manually check the memory network columns, which contribute the most to background meal representations in each reasoning step. One observation is that different columns are selected in each reasoning step. With more reasoning steps, the selected columns become more abstract (less similar to the input meals) compared to the columns selected in the former steps. Another observation is that the selected columns can provide instructive information (e.g. since the burger is wrapped in paper and the description is not detailed in Figure 6 (d), more concrete McDonald's meals are selected), which sheds light on the effectiveness of memory networks for enhancing meal representations.

Conclusion

We have investigated a multi-modal multi-task framework for automatic dietary assessment. Compared to previous dietary assessment methods, we utilize multiple modalities (images, descriptions and user information) as inputs and propose a novel end-to-end framework to give users both meal ratings and comments. In addition, a novel memory network architecture is devised to enable reasoning capabilities over historical meal information. Results on a real-world dataset show that our method is highly competitive, thus providing a tool for automating the advisory process.

Acknowledgments

This research was supported in part by the National Natural Science Foundation of China under Grant no. 61472266. Yue Zhang is the corresponding author.

References

Alwan, A. 2011. *Global status report on noncommunicable diseases 2010.* World Health Organization.

Bahdanau, D., and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Barbu, A., and Bridge, A. 2012. Video in sentences out. *arXiv preprint arXiv:1204.2742*.

Chen, S., and Jin, Q. 2015. Multi-modal dimensional emotion recognition using recurrent neural networks. In *AVEC*, 49–56. ACM.

Cheng, J.; Dong, L.; and Lapata, M. 2016. Long short-term memory-networks for machine reading. *arXiv preprint arXiv:1601.06733*.

Dehais, J.; Anthimopoulos, M.; and Mougiakakou, S. 2015. Dish detection and segmentation for dietary assessment on smartphones. In *ICIAP*, 433–440.

Duong, L.; Cohn, T.; Bird, S.; and Cook, P. 2015. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In *ACL*, 845–850.

Goyal, S.; Liu, Q.; Tajul-Arifin, K.; Awan, W.; Wadhwa, B.; and Liu, Z. 2017. I ate this: A photo-based food journaling system with expert feedback. *arXiv preprint arXiv:1702.05957*.

Hassannejad, H.; Matrella, G.; Ciampolini, P.; De Munari, I.; Mordonini, M.; and Cagnoni, S. 2017. Automatic diet monitoring: a review of computer vision and wearable sensor-based methods. *International Journal of Food Sciences and Nutrition*.

He, X., and Chua, T.-S. 2017. Neural factorization machines for sparse predictive analytics. In *SIGIR*.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. 1735–1780. MIT.

Karpathy, A., and Fei-Fei, L. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3128–3137.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

Kumar, A.; Irsoy, O.; Ondruska, P.; Iyyer, M.; Bradbury, J.; Gulrajani, I.; Zhong, V.; Paulus, R.; and Socher, R. 2016. Ask me anything: Dynamic memory networks for natural language processing. In *International Conference on Machine Learning*, 1378–1387.

Liang, Y., and Li, J. 2017. Deep learning-based food calorie estimation method in dietary assessment. *arXiv preprint arXiv:1706.04062*.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *ACL*. Barcelona, Spain.

Liu, P.; Qiu, X.; and Huang, X. 2016. Recurrent neural network for text classification with multi-task learning. *arXiv preprint arXiv:1605.05101*.

Mansimov, E.; Parisotto, E.; Ba, J. L.; and Salakhutdinov, R. 2015. Generating images from captions with attention. *arXiv preprint arXiv:1511.02793*.

Nam, H., and Han, B. 2016. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4293–4302.

Rebedew, D. 2014. Myfitnesspal. *Fam. Pract. Manage* 22(2):31–31.

Rendle, S. 2010. Factorization machines. In *ICDM*. IEEE.

Roberts, C. K., and Barnard, R. J. 2005. Effects of exercise and diet on chronic disease. *Journal of Applied Physiology*.

Schmidhuber, J. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks* 602–610.

Shan, Y.; Hoens, T. R.; and Jiao, J. 2016. Deep crossing: Web-scale modeling without manually crafted combinatorial features. In *SIGKDD*, 255–262.

Shrivastava, A., and Li, P. 2014. Asymmetric lsh (alsh) for sublinear time maximum inner product search (mips). In *Advances in neural information processing systems*.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sukhbaatar, S.; Weston, J.; Fergus, R.; et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, 2440–2448.

Vaswani, A., and Shazeer, N. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Wang, Z., and Zhang, Y. 2017. Opinion recommendation using neural memory model. *arXiv preprint arXiv:1702.01517*.

Weston, J.; Chopra, S.; and Bordes, A. 2014. Memory networks. *arXiv preprint arXiv:1410.3916*.

Xiong, C.; Merity, S.; and Socher, R. 2016. Dynamic memory networks for visual and textual question answering. In *International Conference on Machine Learning*.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*.

Yang, Y., and Hospedales, T. M. 2016. Trace norm regularised deep multi-task learning. *arXiv preprint arXiv:1606.04038*.

Yang, X.; Molchanov, P.; and Kautz, J. 2016. Multilayer and multimodal fusion of deep neural networks for video classification. In *ACM Multimedia*, 978–987. ACM.

Zhang, Y., and Yang, Q. 2017. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*.