Learning to Respond with Stickers:
A Framework of Unifying Multi-Modality in Multi-Turn Dialog

Shen Gao[*][†]
WICT, Peking University
shengao@pku.edu.cn

Xiuying Chen[*]
WICT, Peking University
xy-chen@pku.edu.cn

Chang Liu
WICT, Peking University
liuchang97@pku.edu.cn

Li Liu
Inception Institute of Artificial
Intelligence
li-liu1985@inceptioniai.org

Dongyan Zhao
WICT, Peking University
zhaody@pku.edu.cn

Rui Yan[‡]
[1] WICT, Peking University
[2] Beijing Academy of Artificial
Intelligence
ruiyan@pku.edu.cn

ABSTRACT

Stickers with vivid and engaging expressions are becoming increasingly popular in online messaging apps, and some works are dedicated to automatically select sticker response by matching text labels of stickers with previous utterances. However, due to their large quantities, it is impractical to require text labels for the all stickers. Hence, in this paper, we propose to recommend an appropriate sticker to user based on multi-turn dialog context history without any external labels. Two main challenges are confronted in this task. One is to learn semantic meaning of stickers without corresponding text labels. Another challenge is to jointly model the candidate sticker with the multi-turn dialog context. To tackle these challenges, we propose a *sticker response selector* (SRS) model. Specifically, SRS first employs a convolutional based sticker image encoder and a self-attention based multi-turn dialog encoder to obtain the representation of stickers and utterances. Next, deep interaction network is proposed to conduct deep matching between the sticker with each utterance in the dialog history. SRS then learns the short-term and long-term dependency between all interaction results by a fusion network to output the the final matching score. To evaluate our proposed method, we collect a large-scale real-world dialog dataset with stickers from one of the most popular online chatting platform. Extensive experiments conducted on this dataset show that our model achieves the state-of-the-art performance for all commonly-used metrics. Experiments also verify the

[*]Equal contribution. Ordering is decided by a coin flip. Work performed during an internship at IIAI.
[†]WICT is the abbreviation of Wangxuan Institute of Computer Technology.
[‡]Corresponding Author: Rui Yan (ruiyan@pku.edu.cn)

effectiveness of each component of SRS. To facilitate further research in sticker selection field, we release this dataset of 340K multi-turn dialog and sticker pairs[1].

1 INTRODUCTION

Images are another important approach for expressing feelings and emotions in addition to using text in communication. In mobile messaging apps, these images can generally be classified into emojis and stickers. Emojis are usually used to help reinforce simple emotions in a text message due to their small size, and their variety is limited. Stickers, on the other hand, can be regarded as an alternative for text messages, which usually include cartoon characters and are of high definition. They can express much more complex and vivid emotion than emojis. Most messaging apps, such as WeChat, Telegram, WhatsApp, and Slack provide convenient ways for users to download stickers for free, or even share self-designed ones. We show a chat window including stickers in Figure 1.

Stickers are becoming more and more popular in online chat. First, sending a sticker with a single click is much more convenient than typing text on the 26-letter keyboard of a small mobile phone screen. Second, there are many implicit or strong emotions that cannot be accurately explained by words but can be captured by stickers with vivid facial expressions and body language. However, the large scale use of stickers means that it is not always straightforward to think of the sticker that best expresses one's feeling according to the current chatting context. Users need to recall all

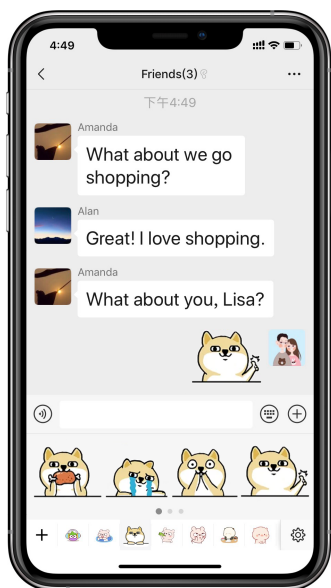[1]https://github.com/gsh199449/stickerchat

Figure 1: An example of stickers in a multi-turn dialog. Sticker response selector automatically selects the proper sticker based on multi-turn dialog history.

the stickers they have collected and selected the appropriate one, which is both difficult and time-consuming.

Consequently, much research has focused on recommending appropriate emojis to users according to the chatting context. Existing works, such as [41], are mostly based on emoji recommendation, where they predict the probable emoji given the contextual information from multi-turn dialog systems. In contrast, other works [5, 6] recommend emojis based on the text and images posted by a user. However, the use of emojis is restricted due to their limited variety and small size, while stickers are more expressive and of a great variety. As for sticker recommendation, existing works such as [23] and apps like Hike or QQ directly match the text typed by the user to the short text tag assigned to each sticker. However, since there are countless ways of expressing the same emotion, it is impossible to capture all variants of an utterance as tags.

In this paper, we address the task of sticker response selection in multi-turn dialog, where an appropriate sticker is recommended based on the dialog history. There are two main challenges in this task: (1) To the best of our knowledge, no existing image recognition methods can model the sticker image, how to capture the semantic meaning of sticker is challenging. (2) Understanding multi-turn dialog history information is crucial for sticker recommendation, and jointly modeling the candidate sticker with multi-turn dialog is challenging. Herein, we propose a novel sticker recommendation model, namely *sticker response selector* (SRS), for sticker response selection in multi-turn dialog. Specifically, SRS first learns representations of dialog context history using a self-attention mechanism and learns the sticker representation by a convolutional network. Next, SRS conducts deep matching between the sticker and each utterance and produces the interaction results for every utterance.

Finally, SRS employs a fusion network which consists of a sub-network fusion RNN and fusion transformer to learn the short and long term dependency of the utterance interaction results. The final matching score is calculated by an interaction function. To evaluate the performance of our model, we propose a large number of multi-turn dialog dataset associated with stickers from one of the popular messaging apps. Extensive experiments conducted on this dataset show that SRS significantly outperforms the state-of-the-art baseline methods in commonly-used metrics.

Our contributions can be summarized as follows:

• We employ a deep interaction network to conduct matching between candidate sticker and each utterance in dialog context.

• We propose a fusion network that can capture the short and long dependency of the interaction results of each utterance simultaneously.

• Experiments conducted on a large-scale real-world dataset[2] show that our model outperforms all baselines, including state-of-the-art models.

2 RELATED WORK

We outline related work on sticker recommendation, visual question answering, visual dialog, and multi-turn response selection.

Sticker recommendation. Most of the previous works emphasize the use of emojis instead of stickers. For example, [5, 6] use a multimodal approach to recommend emojis based on the text and images in an Instagram post. However, emojis are typically used in conjunction with text, while stickers are independent information carriers. What is more, emojis are limited in variety, while there exists an abundance of different stickers. The most similar work to ours is [23], where they generate recommended stickers by first predicting the next message the user is likely to send in the chat, and then substituting it with an appropriate sticker. However, more often than not the implication of the stickers cannot be fully conveyed by text and, in this paper, we focus on directly generating sticker recommendations from dialog history.

Visual question answering. Sticker recommendation involves the representation of and interaction between images and text, which is related to the Visual Question Answering (VQA) task [11, 15, 28, 29, 35]. Specifically, VQA takes an image and a corresponding natural language question as input and outputs the answer. It is a classification problem in which candidate answers are restricted to the most common answers appearing in the dataset and requires deep analysis and understanding of images and questions such as image recognition and object localization [16, 27, 38, 42]. Current models can be classified into three main categories: early fusion models, later fusion models, and external knowledge-based models. One state-of-the-art VQA model is [25], which proposes an architecture, positional self-attention with co-attention, that does not require a recurrent neural network (RNN) for video question answering. [17] proposes an image-question-answer synergistic network, where candidate answers are coarsely scored according to their relevance to the image and question pair in the first stage. Then, answers with a high probability of being correct are re-ranked by synergizing with images and questions.

---

[2]https://github.com/gsh199449/stickerchat

The difference between sticker selection and VQA task is that sticker selection task focus more on multi-turn multimodal interaction between stickers and utterances.

Visual dialog. Visual dialog extends the single turn dialog task [14, 31] in VQA to a multi-turn one, where later questions may be related to former question-answer pairs. To solve this task, [26] transfers knowledge from a pre-trained discriminative network to a generative network with an RNN encoder, using a perceptual loss. [39] combines reinforcement learning and generative adversarial networks (GANs) to generate more human-like responses to questions, where the GAN helps overcome the relative paucity of training data, and the tendency of the typical maximum-likelihood-estimation-based approach to generate overly terse answers. [21] demonstrates a simple symmetric discriminative baseline that can be applied to both predicting an answer as well as predicting a question in the visual dialog.

Unlike VQA and visual dialog tasks, in a sticker recommendation system, the candidates are stickers rather than text.

Multi-turn response selection. Multi-turn response selection [10, 33, 43–45] takes a message and utterances in its previous turns as input and selects a response that is natural and relevant to the whole context. In our task, we also need to take previous multi-turn dialog into consideration. Previous works include [46], which uses an RNN to represent context and response, and measure their relevance. More recently, [40] matches a response with each utterance in the context on multiple levels of granularity, and the vectors are then combined through an RNN. The final matching score is calculated by the hidden states of the RNN. [47] extends this work by considering the matching with dependency information. More recently, [32] proposes a multi-representation fusion network where the representations can be fused into matching at an early stage, an intermediate stage, or at the last stage.

Traditional multi-turn response selection deals with pure natural language processing, while in our task, we also need to obtain a deep understanding of images.

3 DATASET

In this section, we introduce our multi-turn dialog dataset with sticker as response in detail.

3.1 Data Collection

We collect the large-scale multi-turn dialog dataset with stickers from one of the most popular messaging apps. In this app, a large mount of sticker sets are published, and everyone can use the sticker when chatting with a friend or in a chat group. Specifically, we select 20 public chat groups consisting of active members, which are all open groups that everyone can join it without any authorities. The chat history of these groups is collected along with the complete sticker sets. These sticker sets include stickers with similar style. All stickers are resized to a uniform size of 128×128 pixels. We use 20 utterances before the sticker response as the dialog context, and then we filter out irrelevant utterance sentences, such as URL links and attached files. Due to privacy concern, we also filter out user information and anonymize user IDs. To construct negative samples, 9 stickers other than the ground truth sticker are randomly sampled

Table 1: Statistics of Response Selection Dataset.

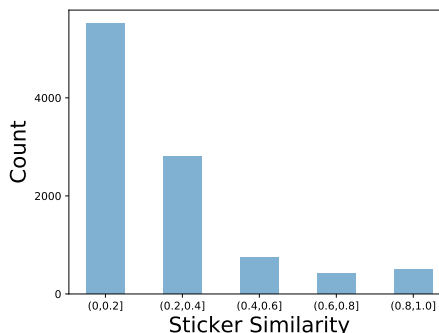|	Train	Valid	Test
# context-stickers pairs	320,168	10,000	10,000
Avg. words of context utterance	7.54	7.50	7.42
Avg. users participate	5.81	5.81	5.79

Figure 2: Similarity distribution among all stickers in test dataset.

from the sticker set. After pre-processing, there are 320,168 context-sticker pairs in the training dataset, 10,000 pairs in the validation, and 10,000 pairs in test datasets respectively. We make sure that there is no overlap between these three datasets. Two examples are shown in Figure 3. We publish this dataset to communities to facilitate further research on dialog response selection task.

3.2 Statistics and Analysis

In total, there are 3,516 sets of sticker which contain 174,695 stickers. The average number of stickers in a sticker set is 49.64. Each context includes 15.5 utterances on average. The average number of users who participate in the dialog context over each dataset is shown in the third row of Table 1.

3.3 Sticker Similarity

Stickers in the same set always share a same style or contain the same cartoon characters. Intuitively, the more similar the candidate stickers are, the more difficult it is to choose the correct sticker from candidates. In other words, the similarity between candidate stickers determines the difficulty of the sticker selection task. To investigate the difficulty of this task, we calculate the average similarity of all the stickers in a specific sticker set by the Structural Similarity Index (SSIM) metric [2, 37]. We first calculate the similarity between the ground truth sticker and each negative sample, then average the similarity scores. The similarity distribution among test data is shown in Figure 2, where the average similarity is 0.258. The examples in Figure 3 are also used to illustrate the similarity of stickers more intuitively, where the left one has a relatively low similarity score, and the right one has a high similarity score.

Case 1 (AVG. 0.125)

- 你好像挺喜欢用这个表情包的呀
- You seem to like using this sticker set very much.
- 是的，app官方的表情包做得很好，深得我心
- Yes, the official sticker set of this app is designed very well and win my heart.
- 我也挺喜欢app官方的表情包
- I like the official sticker sets too.

Case 2 (AVG. 0.702)

- wsSettings里的Header参数是什么意思？怎么用？
- What is the Header parameter in wsSettings? how to use?
- 指的是http的那个吧
- It probably means the parameter of HTTP.
- 呃呃呃，是的...感谢
- Ummm, yes, thanks.
- 哈哈，成功了？
- Haha, have you succeeded?
- 恩恩
- Yes.

Figure 3: Example cases in the dataset with different similarity scores.

4 PROBLEM FORMULATION

Before presenting our approach for sticker response selection in multi-turn dialog, we first introduce our notations and key concepts.

Similar to the multi-turn dialog response selection [40, 47], we assume that there is a multi-turn dialog context $s = \{u_1, \ldots, u_{T_u}\}$ and a candidate sticker set $C = \{c_1, \ldots c_{T_c}\}$, where u_i represents the i-th utterance in the multi-turn dialog. In the i-th utterance $u_i = \{x_1^i, \ldots, x_{T_x^i}^i\}$, x_j^i represents the j-th word in u_i, and T_x^i represents the total number of words in u_i utterance. In dialog context s, c_i represents a sticker image with a binary label y_i, indicating whether c_i is an appropriate response for s. T_u is the utterance number in the dialog context and T_c is the number of candidate stickers. For each candidate set, there is only one ground truth sticker, and the remaining ones are negative samples. Our goal is to learn a ranking model that can produce the correct ranking for each candidate sticker c_i; that is, can select the correct sticker among all the other candidates. For the rest of the paper, we take the i-th candidate sticker c_i as an example to illustrate the details of our model and omit the candidate index i for brevity.

5 SRS MODEL

5.1 Overview

In this section, we propose our *sticker response selector*, abbreviated as SRS. An overview of SRS is shown in Figure 4, which can be split into four main parts:

- *Sticker encoder* is a convolutional neural network (CNN) based image encoding module that learns a sticker representation.
- *Utterance encoder* is a self-attention mechanism based module encoding each utterance u_i in the multi-turn dialog context s.
- *Deep interaction network* module conducts deep matching between each sticker representation and each utterance, and outputs each interaction result.
- *Fusion network* learns the short-term dependency by the fusion RNN and the long-term dependency by the fusion Transformer, and

finally outputs the matching score combining these features using an interaction function.

5.2 Sticker Encoder

Much research has been conducted to alleviate gradient vanishing [19] and reduce computational costs [18] in image modeling tasks. We utilize one of these models, *i.e.*, the Inception-v3 [30] model rather than plain CNN to encode sticker image:

$$O, O_{\text{flat}} = \text{Inception-v3}(c), \quad (1)$$

where c is the sticker image. The sticker representation is $O \in \mathbb{R}^{p \times p \times d}$ which conserves the two-dimensional information of the sticker, and will be used when associating stickers and utterances in §5.4. We use the original image representation output of Inception-v3 $O_{\text{flat}} \in \mathbb{R}^d$ as another sticker representation. However, existing pre-trained CNN networks including Inception-v3 are mostly built on real-world photos. Thus, directly applying the pre-trained networks on stickers cannot speed up the training process. In this dataset, sticker author give each sticker c an emoji tag which denotes the general emotion of the sticker. Hereby, we propose an auxiliary sticker classification task to help the model converge quickly, which uses O_{flat} to predict which emoji is attached to the corresponding sticker. More specifically, we feed O_{flat} into a linear classification layer and then use the cross-entropy loss $\mathcal{L}_s$ as the loss function of this classification task.

5.3 Utterance Encoder

To model the semantic meaning of the dialog context, we learn the representation of each utterance u_i. First, we use an embedding matrix e to map a one-hot representation of each word in each utterance u_i to a high-dimensional vector space. We denote $e(x_j^i)$ as the embedding representation of word x_j^i. From these embedding representations, we use the attentive module from Transformer [34] to model the temporal interactions between the words in an utterance. Attention mechanisms have become an integral part of
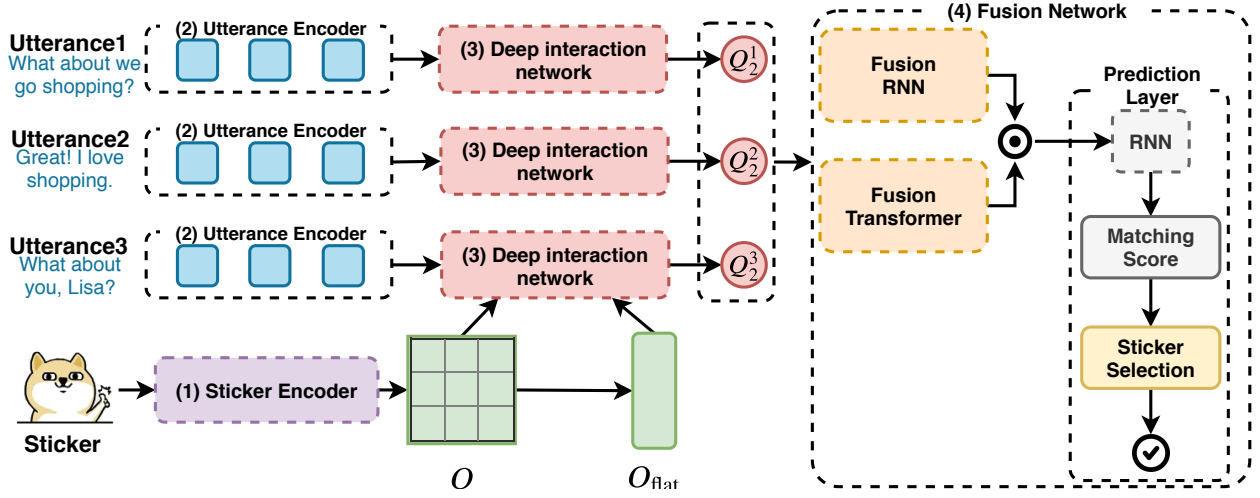
Figure 4: Overview of SRS. We divide our model into four ingredients: (1) *Sticker encoder* learns sticker representation; (2) *Utterance encoder* learns representation of each utterance; (3) *Deep interaction network* conducts deep matching interaction between sticker representation and utterance representation in different levels of granularity. (4) *Fusion network* combines the long-term and short-term dependency feature between interaction results produced by (3).

compelling sequence modeling in various tasks [4, 9, 13, 25]. In our sticker selection task, we also need to let words fully interact with each other to model the dependencies of words without regard to their locations in the input sentence. The attentive module in the Transformer has three inputs: the query Q, the key K and the value V. We use three fully-connected layers with different parameters to project the embedding of dialog context $e(x_j^i)$ into three spaces:

$$Q_j^i = FC(e(x_j^i)), K_j^i = FC(e(x_j^i)), V_j^i = FC(e(x_j^i)). \quad (2)$$

The attentive module then takes each Q_j^i to attend to $K_.^i$, and uses these attention distribution results $\alpha_{j,.}^i \in \mathbb{R}^{T_x^i}$ as weights to gain the weighted sum of V_j^i as shown in Equation 4. Next, we add the original word representations on β_j^i as the residential connection layer, shown in Equation 5:

$$\alpha_{j,k}^i = \frac{\exp\left(Q_j^i \cdot K_k^i\right)}{\sum_{n=1}^{T_x^i} \exp\left(Q_j^i \cdot K_n^i\right)}, \quad (3)$$

$$\beta_j^i = \sum_{k=1}^{T_x^i} \alpha_{j,k}^i \cdot V_k^i, \quad (4)$$

$$\hat{h}_j^i = \text{Dropout}\left(e(x_j^i) + \beta_j^i\right), \quad (5)$$

where $\alpha_{j,k}^i$ denotes the attention weight between j-th word to k-th word in i-th utterance. To prevent vanishing or exploding of gradients, a layer normalization operation [24] is also applied on the output of the feed-forward layer as shown in Equation 6:

$$h_j^i = \text{norm}\left(\max(0, \hat{h}_j^i \cdot W_1 + b_1) \cdot W_2 + b_2\right), \quad (6)$$

where W_1, W_2, b_1, b_2 are all trainable parameters. h_j^i denotes the hidden state of j-th word in the Transformer for the i-th utterance.

5.4 Deep Interaction Network

Now that we have the representation of the sticker and each utterance, we can conduct a deep matching between these components. On one hand, there are some emotional words in dialog context history that match the expression of the stickers such as "happy" or "sad". On the other hand, specific part of the sticker can also match these corresponding words such as dancing limbs or streaming eyes. Hence, we employ a bi-directional attention mechanism between a sticker and each utterance, that is, from utterance to sticker and from sticker to utterance, to analyze the cross-dependency between the two components. The interaction is illustrated in Figure 5.

We take the i-th utterance as an example and omit the index i for brevity. The two directed attentions are derived from a shared relation matrix, $M \in \mathbb{R}^{(p^2) \times T_u}$, calculated by sticker representation $O \in \mathbb{R}^{p \times p \times d}$ and utterance representation $h \in \mathbb{R}^{T_u \times d}$. The score $M_{kj} \in \mathbb{R}$ in the relation matrix M indicates the relation between the k-th sticker representation unit O_k, $k \in [1, p^2]$ and the j-th word h_j, $j \in [1, T_u]$ and is computed as:

$$M_{kj} = \sigma(O_k, h_j), \quad \sigma(x, y) = w^\mathsf{T}[x \oplus y \oplus (x \otimes y)], \quad (7)$$

where σ is a trainable scalar function that encodes the relation between two input vectors. $\oplus$ denotes a concatenation operation and $\otimes$ is the element-wise multiplication.

Next, a max pooling operation is conducted on M, i.e., let $\tau_j^u = \max(M_{\cdot j}) \in \mathbb{R}$ represent the attention weight on the j-th utterance word by the sticker representation, corresponding to the "utterance-wise attention". This attention learns to assign high weights to the important words that are closely related to sticker. We then obtain the weighted sum of hidden states as "**sticker-aware utterance representation**" l:
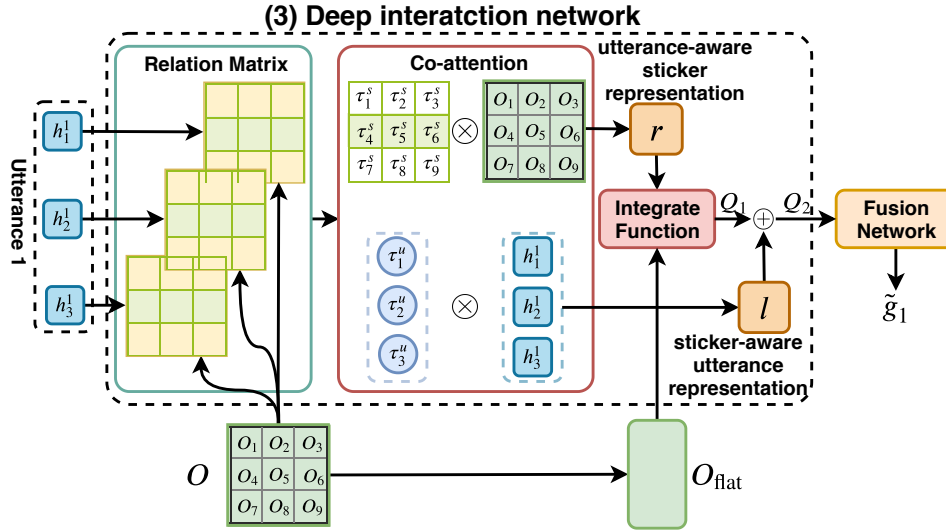
$$l = \sum_j^{T_u} \tau_j^u h_j. \quad (8)$$

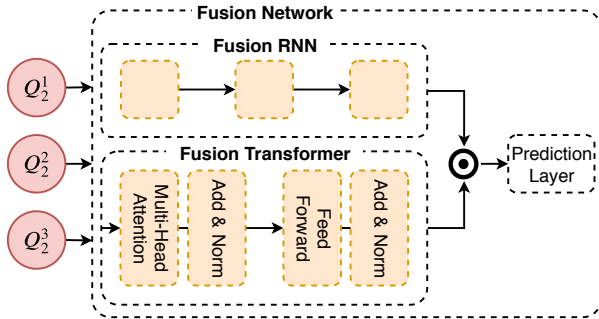Figure 5: Framework of deep interaction network.

Figure 6: Framework of fusion network.

Similarly, sticker-wise attention learns which part of a sticker is most relevant to the utterance. Let $\tau_k^s = \max(M_{k:}) \in \mathbb{R}$ represent the attention weight on the k-th unit of the sticker representation. We use this to obtain the weighted sum of O_k, *i.e.,*the "**utterance-aware sticker representation**" r:

$$r = \sum_k^{p^2} \tau_k^s O_k. \tag{9}$$

After obtaining the two outputs from the co-attention module, we combine the sticker and utterance representations and finally get the ranking result. We first integrate the utterance-aware sticker representation r with the original sticker representation O_{flat} using an **integrate function**, named IF:

$$Q_1 = IF(O_{\text{flat}}, r), \quad IF(x, y) = FC(x \oplus y \oplus (x \otimes y) \oplus (x + y)). \tag{10}$$

We add the sticker-aware utterance representation l into Q_1 together and then apply a fully-connected layer:

$$Q_2 = FC(Q_1 \oplus l). \tag{11}$$

5.5 Fusion Network

Up till now, we have obtained the interaction result between each utterance and the candidate sticker. Here we again include the utterance index i where Q_2 now becomes Q_2^i. Since the utterances in a multi-turn dialog context are in chronological order, we employ a **Fusion RNN** and a **Fusion Transformer** to model the short-term and long-term interaction between utterance $\{Q_2^1, \ldots, Q_2^{T_u}\}$.

5.5.1 Fusion RNN. Fusion RNN first reads the interaction results for each utterance $\{Q_2^1, \ldots, Q_2^{T_u}\}$ and then transforms into a sequence of hidden states. In this paper, we employ the gated recurrent unit (GRU) [7] as the cell of fusion RNN, which is popular in sequential modeling [12, 40]:

$$g_i = \text{RNN}(Q_2^i, g_{i-1}), \tag{12}$$

where g_i is the hidden state of the fusion RNN. Finally, we obtain the sequence of hidden states $\{g_1, \ldots, g_{T_u}\}$. One can replace GRU with similar algorithms such as LSTM [20]. We leave the study as future work.

5.5.2 Fusion Transformer. To model the long-term dependency and capture the salience utterance from the context, we employ the self-attention mechanism introduced in Equation 3-6. Concretely, given $\{Q_2^1, \ldots, Q_2^{T_u}\}$, we first employ three linear projection layers with different parameters to project the input sequence into three different spaces:

$$Q^i = FC(Q_2^i), \quad \mathcal{K}^i = FC(Q_2^i), \quad \mathcal{V}^i = FC(Q_2^i). \tag{13}$$

Then we feed these three matrices into the self-attention algorithm illustrated in Equation 3-6. Finally, we obtain the long-term interaction result $\{\hat{g}_1, \ldots, \hat{g}_{T_u}\}$.

5.5.3 Prediction Layer. To combine the interaction representation generated by fusion RNN and fusion Transformer, we employ the

SUMULTI function proposed by [36] to combine these representations, which has been proven effective in various tasks:

$$\overline{g}_i = \text{ReLU}(\mathcal{W}^s \begin{bmatrix} (\hat{g}_i - g_i) \odot (\hat{g}_i - g_i) \\ \hat{g}_i \odot g_i \end{bmatrix} + \mathbf{b}^s). \quad (14)$$

The new interaction sequence $\{\overline{g}_1, \ldots, \overline{g}_{T_u}\}$ is then boiled down to a matching vector $\tilde{g}_{T_u}$ by another GRU-based RNN:

$$\tilde{g}_i = \text{RNN}(\tilde{g}_{i-1}, \overline{g}_i). \quad (15)$$

We use the final hidden state $\tilde{g}_{T_u}$ as the representation of the overall interaction result between the whole utterance context and the candidate sticker. Finally, we apply a fully-connected layer to produce the matching score $\hat{y}$ of the candidate sticker:

$$\hat{y} = FC(\tilde{g}_{T_u}), \quad (16)$$

where $\hat{y} \in (0, 1)$ is the matching score of the candidate sticker.

5.6 Learning

Recall that we have a candidate sticker set $C = \{c_1, \ldots c_{T_c}\}$ which contains multiple negative samples and one ground truth sticker. We use hinge loss as our objective function:

$$\mathcal{L}_r = \sum^N \max\left(0, \hat{y}_{\text{negative}} - \hat{y}_{\text{positive}} + \text{margin}\right), \quad (17)$$

where $\hat{y}_{\text{negative}}$ and $\hat{y}_{\text{positive}}$ corresponds to the predicted labels of the negative sample and ground truth sticker, respectively. The margin is the margin rescaling in hinge loss. The gradient descent method is employed to update all the parameters in our model to minimize this loss function.

6 EXPERIMENTAL SETUP

6.1 Research Questions

We list four research questions that guide the experiments:
• **RQ1** (See § 7.1): What is the overall performance of SRS compared with all baselines?
• **RQ2** (See § 7.2): What is the effect of each module in SRS?
• **RQ3** (See § 7.3): How does the performance change when the number of utterances changes?
• **RQ4** (See § 7.4): Can co-attention mechanism successfully capture the salient part on the sticker image and the important words in dialog context?
• **RQ5** (See § 7.5): What is the influence of the similarity between candidate stickers?
• **RQ6** (See § 7.6): What is the influence of the parameter settings?

6.2 Comparison Methods

We first conduct an ablation study to prove the effectiveness of each component in SRS as shown in Table 2. Specifically, we remove each key part of our SRS to create ablation models and then evaluate the performance of these models.

Next, to evaluate the performance of our model, we compare it with the following baselines. Note that, since no existing models can be directly applied to our task, we adapt VQA and multi-turn response selection models to the sticker response selection task.
(1) **Synergistic**: [17] devises a novel synergistic network on VQA task. First, candidate answers are coarsely scored according to their relevance to the image-question pair. Afterward, answers with

Table 2: Ablation models for comparison.

Acronym	Gloss
SRS w/o pretrain	SRS w/o pre-trained Inception-v3 model
SRS w/o Classify	SRS w/o emoji classification task
SRS w/o DIN	SRS w/o **D**eep **I**nteraction **N**etwork
SRS w/o FR	SRS w/o **F**usion **R**NN

high probabilities of being correct are re-ranked by synergizing with image and question. This model achieves the state-of-the-art performance on the Visual Dialog v1.0 dataset [8].
(2) **PSAC**: [25] proposes the positional self-attention with co-attention architecture on VQA task, which does not require RNNs for video question answering. We replace the output probability on the vocabulary size with the probability on candidate sticker set.
(3) **SMN**: [40] proposes a sequential matching network to address response selection for the multi-turn conversation problem. SMN first matches a response with each utterance in the context. Then vectors are accumulated in chronological order through an RNN. The final matching score is calculated with RNN.
(4) **DAM**: [47] extends the transformer model [34] to the multi-turn response selection task, where representations of text segments are constructed using stacked self-attention. Then, truly matched segment pairs are extracted across context and response.
(5) **MRFN**: [32] proposes a multi-representation fusion network, where the representations can be fused into matching at an early stage, at the intermediate stage or the last stage. This is the state-of-the-art model on the multi-turn response selection task.

For the three baselines above, we replace the candidate embedding RNN network with the image encoding CNN network Inception-v3, as used in our model. This network is initialized using a pre-trained model[3] for all baselines and SRS.

6.3 Evaluation Metrics

Following [32, 47], we employ recall at position k in n candidates $R_n@k$ as an evaluation metric, which measures if the positive response is ranked in the top k positions of n candidates. Following [47], we also employ mean average precision (MAP) [3] as an evaluation metric. The statistical significance of differences observed between the performance of two runs is tested using a two-tailed paired t-test and is denoted using ▲ (or ▼) for strong significance at $\alpha = 0.01$.

6.4 Implementation Details

We implement our experiments using TensorFlow [1] on an NVIDIA P100 GPU. If the number of words in an utterance is less than 30, we pad zeros, otherwise, the first 30 words are kept. The word embedding dimension is set to 100 and the number of hidden units is 100. The batch size is set to 32. 9 negative samples are randomly sampled from the sticker set containing the ground truth sticker, and we finally obtain 10 candidate stickers for the model to select. We use Adam optimizer [22] as our optimizing algorithm, and the learning rate is 1×10^{-4}.

---

[3]https://github.com/tensorflow/models/tree/master/research/slim

Table 3: RQ1: Automatic evaluation comparison. Significant differences are with respect to MRFN.

	MAP	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
Visual Q&A methods				
Synergistic	0.593	0.438	0.569	0.798
PSAC	0.662	0.533	0.641	0.836
Multi-turn response selection methods				
SMN	0.524	0.357	0.488	0.737
DAM	0.620	0.474	0.601	0.813
MRFN	0.684	0.557	0.672	0.853
SRS	**0.709**	**0.590**▲	**0.703**▲	**0.872**

Table 4: RQ2: Evaluation of different ablation models.

	MAP	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
SRS w/o pretrain	0.650	0.510	0.641	0.833
SRS w/o Classify	0.707	0.588	0.700	0.871
SRS w/o DIN	0.680	0.552	0.669	0.854
SRS w/o FR	0.677	0.551	0.663	0.863
SRS	**0.709**	**0.590**	**0.703**	**0.872**

7 EXPERIMENTAL RESULT

7.1 Overall Performance

For research question **RQ1**, we examine the performance of our model and baselines in terms of each evaluation metric, as shown in Table 3. First, the performance of the multi-turn response selection models is generally consistent with their performances on text response selection datasets. SMN [40], an earlier work on multi-turn response selection task with a simple structure, obtains the worst performance on both sticker response and text response selection. DAM [47] improves the SMN model and gets the second best performance. MRFN [32] is the state-of-the-art text response selection model and achieves the best performance among baselines in our task as well. Second, VQA models perform generally worse than multi-turn response selection models, since the interaction between multi-turn utterances and sticker is important, which is not taken into account by VQA models. Finally, SRS achieves the best performance with 3.36%, 5.92% and 3.72% improvements in MAP, $R_{10}@1$ and $R_{10}@2$ respectively, over the state-of-the-art multi-turn selection model, *i.e.*,MRFN, and with 6.80%, 10.69% and 8.74% significant increases over the state-of-the-art visual dialog model, PSAC. This proves the superiority of our model.

7.2 Ablation Study

For research question **RQ2**, we conduct ablation tests on the use of the pre-trained Inception-v3 model, the sticker classification loss, the deep interaction network and the fusion RNN respectively. The evaluation results are shown in Table 4. The performances of all ablation models are worse than that of SRS under all metrics, which demonstrates the necessity of each component in SRS. We also find that the sticker classification makes the least contribution to

the overall performance. But this additional task can speed up the training process, and help our model to converge quickly. We use 19 hours to train the SRS until convergence, and we use 30 hours for training SRS w/o Classify. The fusion RNN brings a significant contribution, improving the MAP and $R_{10}@1$ scores by 4.43% and 7.08%, respectively. Besides, the deep interaction network also plays an important part. Without this module, the interaction between the sticker and utterance are hindered, leading to a 6.88% drop in $R_{10}@1$.

7.3 Analysis of Number of Utterances

For research question **RQ3**, in addition to comparing with various baselines, we also evaluate our model when reading different number of utterances to study how the performance relates to number of context turns.

Figure 7 shows how the performance of the SRS changes with respect to different numbers of utterances turns. We observe a similar trend for SRS on the first three evaluation metrics *MAP*, $R_{10}@1$ and $R_{10}@2$: they first increase until the utterance number reaches 15, and then fluctuate as the utterance number continues to increase. There are two possible reasons for this phenomena. The first reason might be that, when the information in the utterances is limited, the model can capture the features well, and thus when the amount of information increases, the performance gets better. However, the capacity of the model is limited, and when the amount of information reaches its upper bound, it gets confused by this overwhelming information. The second reason might be of the usefulness of utterance context. Utterances that occur too early before the sticker response may be irrelevant to the sticker and bring unnecessary noise. As for the last metric, the above observations do not preserve. The $R_{10}@5$ scores fluctuate when the utterance number is below 15, and drop when the utterance number increases. The reason might be that $R_{10}@5$ is not a strict metric, and it is easy to collect this right sticker in the set of half of the whole candidates. Thus, the growth of the information given to SRS does not help it perform better but the noise it brings harms the performance. On the other hand, though the number of utterances changes from 3 to 20, the overall performance of SRS generally remains at a high level, which proves the robustness of our model.

7.4 Analysis of Attention Distribution in Interaction Process

Next, we turn to address **RQ4**. We also show three cases with the dialog context in Figure 8. There are four stickers under each dialog context, one is the selected sticker by our model and other three stickers are random selected candidate stickers. As a main component of SRS, the deep interaction network comprises a bidirectional attention mechanism between the utterance and the sticker, where each word in the utterance and each unit in the sticker representation have a similarity score in the co-attention matrix. To visualize the sticker selection process and to demonstrate the interpretability of SRS, we visualize the sticker-wise attention τ^s (Equation 9) on the original sticker image and show some examples in Figure 8. The lighter the area is, the higher attention it gets.

Facial expressions are an important part in sticker images. Hence, we select several stickers with vivid facial expression in Figure 8.
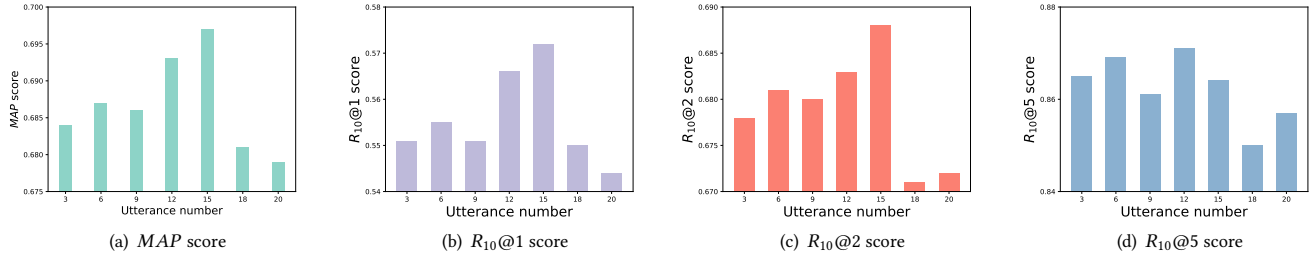
(a) MAP score   (b) $R_{10}@1$ score   (c) $R_{10}@2$ score   (d) $R_{10}@5$ score

Figure 7: Performance of SRS on all metrics when reading different number of utterances.
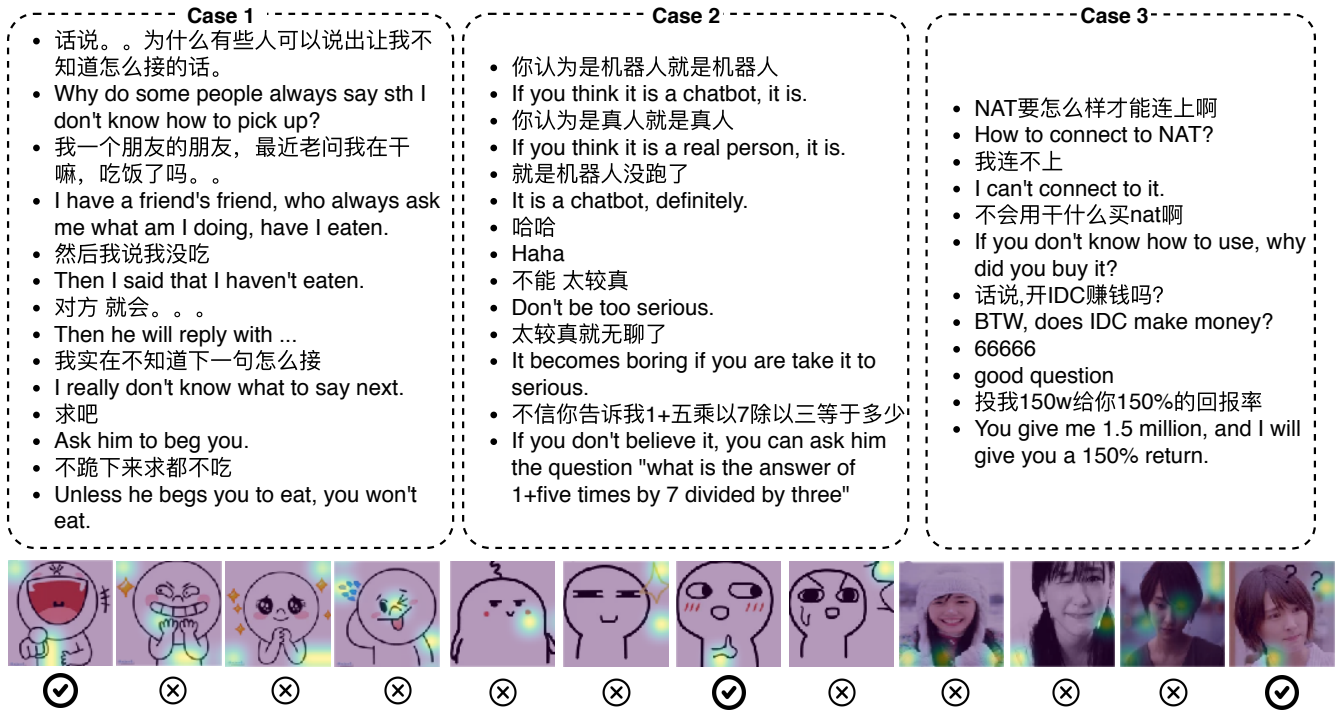

Figure 8: Examples of sticker selection results produced by SRS. We show the selected sticker and three random selected candidate stickers with the attention heat map. The lighter the area on image is, the higher attention weight it gets.

Take forth sticker in Case 1 for example where the character has a wink eye and a smiling mouth. The highlights are accurately placed on the character's eye, indicating that the representation of this sticker is highly dependent on this part. Another example is the last sticker of Case 3, there is two question marks on the top right corner of the sticker image which indicates that the girl is very suspicious of this. In addition to facial expression, the characters gestures can also represent emotions. Take the third sticker in Case 2 for example, the character in this sticker gives a thumbs up representing support and we can find that the attention lies on his hand, indicating that the model learns the key point of his body language.

Furthermore, we randomly select three utterances from the test dataset, and we also visualize the attention distribution over the

words in a utterance, as shown in Figure 9. We use the weight τ_j^u for the j-th word (calculated in Equation 8) as the attention weight. We can find that the attention module always gives a higher attention weight on the salience word, such as the "easy method", "make a lot of money" and "use Chine Mobile".

7.5 Influence of Similarity between Candidates

In this section, we turn to **RQ5** to investigate the influence of the similarities between candidates. The candidate stickers are sampled from the same set, and stickers in a set usually have a similar style. Thus, it is natural to ask: Can our model identify the correct sticker from a set of similar candidates? What is the influence of the similarity between candidate stickers? Hence, we use the Structural Similarity Index (SSIM) metric [2, 37] to calculate the
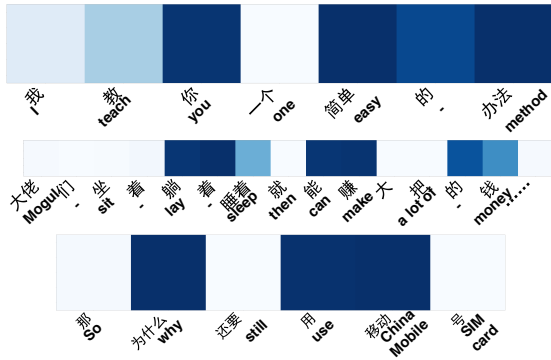
Figure 9: Examples of the attention weights of the dialog utterance. We translate Chinese to English word by word. The darker the area is, the higher weight the word gets.
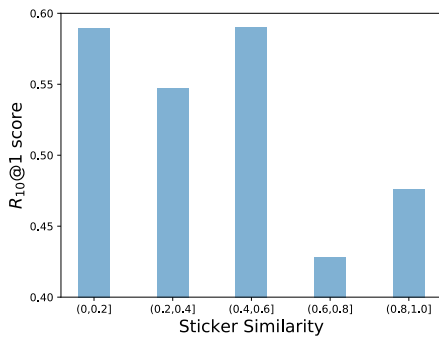

Figure 10: Performance of SRS on groups of different candidate similarity.

Figure 11: Performance of SRS with different parameter settings.

average similarity among all candidates in a test sample and then aggregate all test samples into five groups according to their average similarities. We calculate the $R_{10}@1$ of each group of samples, as shown in Figure 10. The x-axis is the average similarity between candidate stickers and the y-axis is the $R_{10}@1$ score.

Not surprisingly, SRS gains the best performance when the average similarity of the candidate group is low and its performance drops as similarity increases. However, we can also see that, though similarity varies from minimum to maximum, the overall performance can overall stay at high level. $R_{10}@1$ scores of all five groups are above 0.42, and the highest score reaches 0.59. That is, our model is highly robust and can keep giving reasonable sticker responses.

7.6 Robustness of Parameter Setting

Finally, we turn to address **RQ6** to investigate the robustness of parameter setting. We train our model in different parameter setting as shown in Figure 11. The hidden size of the RNN, CNN and the dense layer in our model is tuned from 50 to 250, and we use the MAP and $R_n@k$ to evaluate each model. As the hidden size grows larger from 50 to 100, the performance rises too. The increment of hidden size improves the MAP and $R_{10}@1$ scores by 0.4% and 1.0%. When the hidden size continuously goes larger from 100 to 250, the performance is declined slightly. The increment of hidden size leads
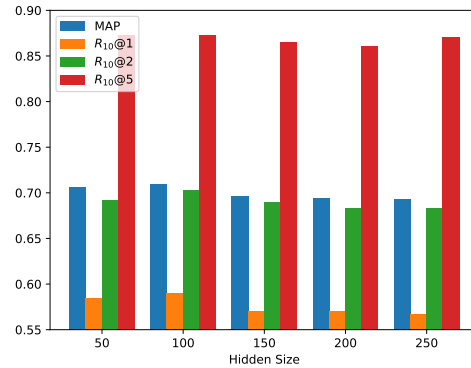
to a 2.2% and 3.9% drop in terms of MAP and $R_{10}@1$ respectively. Nonetheless, we can find that each metric maintained at a stable interval, which demonstrates that our SRS is robust in terms of the parameter size.

8 CONCLUSION

In this paper, we propose the task of multi-turn sticker response selection, which recommends an appropriate sticker based on multi-turn dialog context history without relying on external knowledge. To tackle this task, we proposed the *sticker response selector* (SRS). Specifically, SRS first learns the representation of each utterance using a self-attention mechanism, and learns sticker representation by CNN. Next, a deep interaction network is employed to fully model the dependency between the sticker and utterances. The deep interaction network consists of a co-attention matrix that calculates the attention between each word in an utterance and each unit in a sticker representation. Then, a bi-directional attention is used to obtain utterance-aware sticker representation and sticker-aware utterance representations. Finally, a fusion network models the short-term and long-term relationship between interaction results, and a fully-connected layer is applied to obtain the final selection result. Our model outperforms state-of-the-art methods in all metrics and the experimental results also demonstrate the robustness of our model on datasets with different similarity between candidate stickers. In the near future, we aim to propose a personalized sticker response selection system.

REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: a system for large-scale machine learning.. In *OSDI*, Vol. 16. 265–283.

[2] Alireza Avanaki. 2008. Exact histogram specification optimized for structural similarity. *arXiv preprint arXiv:0901.0065* (2008).

[3] Ricardo Baeza-Yates, Berthier de Araújo Neto Ribeiro, et al. 2011. *Modern information retrieval*. New York: ACM Press; Harlow, England: Addison-Wesley,.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

[5] Francesco Barbieri, Miguel Ballesteros, Francesco Ronzano, and Horacio Saggion. 2018. Multimodal Emoji Prediction. In *NAACL*. Association for Computational Linguistics, New Orleans, Louisiana, 679–686.

[6] Francesco Barbieri, Miguel Ballesteros, and Horacio Saggion. 2017. Are Emojis Predictable?. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics, Valencia, Spain, 105–111.

[7] Junyoung Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. In *NIPS Workshop*.

[8] A. Das, S. Kottur, K. Gupta, A. Singh, D. Yadav, J. M. F. Moura, D. Parikh, and D. Batra. 2017. Visual Dialog. In *CVPR*. 1080–1089.

[9] Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical Neural Story Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 889–898.

[10] Jiazhan Feng, Chongyang Tao, Wei Wu, Yansong Feng, Dongyan Zhao, and Rui Yan. 2019. Learning a Matching Model with Co-teaching for Multi-turn Response Selection in Retrieval-based Dialogue Systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 3805–3815.

[11] Peng Gao, Haoxuan You, Zhanpeng Zhang, Xiaogang Wang, and Hongsheng Li. 2019. Multi-Modality Latent Interaction Network for Visual Question Answering. In *ICCV*.

[12] Shen Gao, Xiuying Chen, Piji Li, Zhangming Chan, Dongyan Zhao, and Rui Yan. 2019. How to Write Summaries with Patterns? Learning towards Abstractive Summarization through Prototype Editing. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 3741–3751.

[13] Shen Gao, Xiuying Chen, Piji Li, Zhaochun Ren, Lidong Bing, Dongyan Zhao, and Rui Yan. 2019. Abstractive Text Summarization by Incorporating Reader Comments. In *AAAI*. 6399–6406.

[14] Shen Gao, Zhaochun Ren, Yihong Zhao, Dongyan Zhao, Dawei Yin, and Rui Yan. 2019. Product-Aware Answer Generation in E-Commerce Question-Answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. Association for Computing Machinery, New York, NY, USA, 429–437.

[15] Ankit Goyal, Jian Wang, and Jia Deng. 2018. Think Visually: Question Answering through Virtual Imagery. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 2598–2608.

[16] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *CVPR*. 6904–6913.

[17] D. Guo, C. Xu, and D. Tao. 2019. Image-Question-Answer Synergistic Network for Visual Dialog. In *CVPR '19*. 10426–10435.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*. 1026–1034.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.

[20] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[21] Unnat Jain, Svetlana Lazebnik, and Alexander G Schwing. 2018. Two can play this game: visual dialog with discriminative question generation and answering. In *ICCV*. 5754–5763.

[22] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*, Vol. abs/1412.6980.

[23] Abhishek Laddha, Mohamed Hanoosh, and Debdoot Mukherjee. 2019. Understanding Chat Messages for Sticker Recommendation in Hike Messenger. *arXiv:1902.02704* (2019).

[24] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[25] Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. 2019. Beyond RNNs: Positional Self-Attention with Co-Attention for Video Question Answering. In *AAAI*.

[26] Jiasen Lu, Anitha Kannan, Jianwei Yang, Devi Parikh, and Dhruv Batra. 2017. Best of both worlds: Transferring knowledge from discriminative learning to a generative visual dialog model. In *NIPS*. 314–324.

[27] Mateusz Malinowski, Marcus Rohrbach, and Mario Fritz. 2015. Ask your neurons: A neural-based approach to answering questions about images. In *ICCV*. 1–9.

[28] H. Noh, T. Kim, J. Mun, and B. Han. 2019. Transfer Learning via Unsupervised Task Discovery for Visual Question Answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8377–8386.

[29] F. Sha, H. Hu, and W. Chao. 2018. Cross-Dataset Adaptation for Visual Question Answering. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5716–5725.

[30] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *CVPR*. 2818–2826.

[31] Chongyang Tao, Shen Gao, Mingyue Shang, Wei Wu, Dongyan Zhao, and Rui Yan. 2018. Get The Point of My Utterance! Learning Towards Effective Responses with Multi-Head Attention Mechanism. In *IJCAI*.

[32] Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. Multi-Representation Fusion Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. In *WSDM*. ACM, 267–275.

[33] Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. One Time of Interaction May Not Be Enough: Go Deep with an Interaction-over-Interaction Network for Response Selection in Dialogues. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 1–11.

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*. 5998–6008.

[35] Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Explicit Knowledge-based Reasoning for Visual Question Answering. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 1290–1296.

[36] Shuohang Wang and Jing Jiang. 2017. A Compare-Aggregate Model for Matching Text Sequences. *ICLR* (2017).

[37] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.

[38] Qi Wu, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2016. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *CVPR*. 4622–4630.

[39] Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton van den Hengel. 2018. Are you talking to me? reasoned visual dialog generation through adversarial learning. In *ICCV*. 6106–6115.

[40] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2017. Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-Based Chatbots. In *ACL*. Association for Computational Linguistics, Vancouver, Canada, 496–505.

[41] Ruobing Xie, Zhiyuan Liu, Rui Yan, and Maosong Sun. 2016. Neural emoji recommendation in dialogue systems. *arXiv preprint arXiv:1612.04609* (2016).

[42] Caiming Xiong, Stephen Merity, and Richard Socher. 2016. Dynamic memory networks for visual and textual question answering. In *ICML*. 2397–2406.

[43] Rui Yan, Yiping Song, and Hua Wu. 2016. Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '16)*. Association for Computing Machinery, New York, NY, USA, 55–64.

[44] Rui Yan and Dongyan Zhao. 2018. Coupled Context Modeling for Deep Chit-Chat: Towards Conversations between Human and Computer. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18)*. Association for Computing Machinery, New York, NY, USA, 2574–2583.

[45] Rui Yan, Dongyan Zhao, and Weinan E. 2017. Joint Learning of Response Ranking and Next Utterance Suggestion in Human-Computer Conversation System. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 685–694.

[46] Xiangyang Zhou, Daxiang Dong, Hua Wu, Shiqi Zhao, Dianhai Yu, Hao Tian, Xuan Liu, and Rui Yan. 2016. Multi-view Response Selection for Human-Computer Conversation. In *EMNLP '16*. Association for Computational Linguistics, Austin, Texas, 372–381.

[47] Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *ACL*, Vol. 1. 1118–1127.