Reactive Multi-Stage Feature Fusion for Multimodal Dialogue Modeling

Yi-Ting Yeh, Tzu-Chuan Lin, Hsiao-Hua Cheng, Yi-Hsuan Deng, Shang-Yu Su, Yun-Nung Chen

National Taiwan University, Taiwan

{r07922064,b04705003,b03902024,b04902013,f05921117}@csie.ntu.edu.tw y.v.chen@ieee.org

Abstract

Visual question answering and visual dialogue tasks have been increasingly studied in the multimodal field towards more practical real-world scenarios. A more challenging task, audio visual scene-aware dialogue (AVSD), is proposed to further advance the technologies that connect audio, vision, and language, which introduces temporal video information and dialogue interactions between a questioner and an answerer. This paper proposes an intuitive mechanism that fuses features and attention in multiple stages in order to well integrate multimodal features, and the results demonstrate its capability in the experiments. Also, we apply several state-of-the-art models in other tasks to the AVSD task, and further analyze their generalization across different tasks.

Introduction

Nowadays, many real-world problems require to integrate information from multiple sources. Sometimes such problems involve distinct modalities, such as vision, language and audio. Therefore, many AI researchers are trying to build models that can deal with tasks using different modalities, and multimodal applications have become an increasing popular research topic. For example, visual question answering (VQA) (Antol et al. 2015) is the task that aims at testing whether an AI model can successfully summarize an image, text description of that image and then generate a correct response. The visual dialog task focuses on examining whether a system can interact with human via conversations (Das et al. 2017), where in a conversation, the system is expected to answer questions correctly given an input image. However, it is difficult to well converse with users when only accessing a single image without audio and dynamic scenes.

Motivated by the need for scene-aware dialogue systems, audio visual scene-aware dialogue (AVSD) is recently proposed by Alamri et al., providing more dynamic scenes with video modal rather than static images in VQA or Visual dialogue tasks. With richer information, machines can be trained to carry on a conversation with users about objects and events around them. To tackle this problem, this paper focuses on 1) better encoding multimodal features and 2) better decoding the responses with consideration of the encoded information. We propose an reactive encoder that is

capable of fusing multimodal features and paying attention correctly, and a decoder that investigates different decoding mechanisms for better generating the dialogue responses. In this paper, our main contributions are 3-fold:

- This paper proposes a simple but effective way to fuse different modalities by the 1×1 convolution followed by a weighted sum operation.
- The proposed multi-stage fusion mechanism can encourage the model to thoroughly understand the question.
- This paper first attempts to generalize top-down attention proposed by Anderson et al. and investigates different attentional decoding methods in a principled way.

Task Description

The task is audio visual scene-aware dialogue (AVSD), which tests the capability of dialogue responses given audio, visual and dialogue contexts. The process of data collection is illustrated in the left part of Figure 1, where the collected dataset contains 11,156 visual dialogues. Each dialogue contains a video and dialogue texts, where there are pre-extracted feature modalities using VGGish (Hershey et al. 2017) (**vggish**) and I3D models (Carreira and Zisserman 2017) (**i3d**) for the video, and the dialogue texts contain a video caption, a summary written by the questioner after 10 rounds Q/A and dialogue history, denoted as **caption**, **summary** and **dialogue** respectively. Note that our model in the experiments only considers *answer* parts as our **dialogue** features. Our goal is to play the role of an answerer, who can reply the reasonable answer to the questioner illustrated in the right part of Figure 1.

Proposed Approach

Our system can be viewed as an encoder-decoder model, which is commonly used in conversation modeling (Sutskever, Vinyals, and Le 2014). We first explain the feature selection procedure, describe the basic fusion model, and then detail the novel design for the proposed model. In the encoder, we apply basic fusion, multi-stage fusion, and 1x1 convolution fusion methods. In the decoder, we apply an original decoder, an attention decoder, and a top-down attention LSTM decoder. The detail is described below.
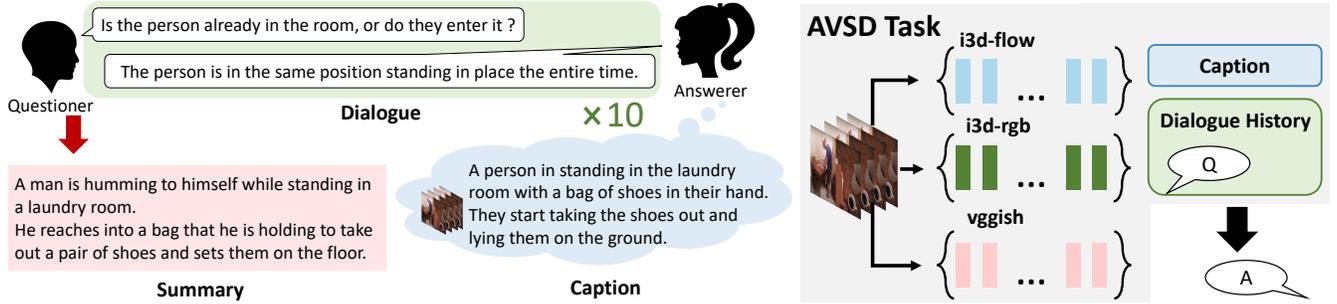
Figure 1: The illustration of the collected data and the audio visual scene-aware dialogue (AVSD) task.

Type	Sentence
Caption	Man walks over to laptop and throws towel over his shoulder. He sits down and wipes and scratches his face with his hands and begins staring at the laptop.
Summary	A man walks into a room and sits down while looking at a laptop on the ***floor***.

Table 1: Misleading **summary** example.

Feature Selection

Considering that multiple modalities are involved in this task, it is important to choose proper and useful features for it. However, some features are too noisy to model and may result in ambiguity. In our proposed model, we decide to ignore **i3d** and **summary** for our feature set.

There are several reasons to drop **i3d**. The main reason is that **i3d** has too much information and thus contains many noises, which may hurt the stability of our model. We assume that the useful information contained in **i3d** can be obtained by combining **vggish** and texts. Because the audio tends to attract attention of people and be the key in the video, **vggish** provides more compact information and texts supply detailed description. In our experiments, we also found that removing **i3d** does not lead to a huge degradation of performance, implying that in fact we can answer most questions without this feature set in our model setup.

The reason of removing **summary** in our feature set is the concern about misleading and impracticality. We observe that **caption** usually contains more concise description and more rich information than **summary**, so our model only concentrates on **caption** to better answer questions. Another consideration is about how **summary** is generated. It is also more reasonable and practical to remove **summary** from our feature set, considering that the summary is generated by the questioner who does not see the video. For example, Table 1 shows a misleading **summary** example that contains incorrect information.

Feature Encoder

After removing noisy features, our model utilizes several sequential features in different forms for the task. Let $x^i = \{x_1^i; \ldots; x_{T_i}^i\}$ represent the i-th feature, which can contain both video features and text word embeddings. For text-based features, *Glove* is applied to obtain word embeddings. Then for each temporal sequence of features, an uni-directional GRU encoder is used to encode all sequential features into a common space $\mathbb{R}^d$, producing $z^i = \{z_1^i; \ldots; z_{T_i}^i\}$, where $z_t^i \in \mathbb{R}^d$ and can be the representation of the whole video or the dialogue history. The illustration can be found in Figure 1.

Basic Fusion Model

In our basic fusion model, we use the last hidden states $\{z_{T_1}^1; \ldots; z_{T_n}^n\}$ for each modality as the encoded representation, and fuse different modalities into a global multimodal representation, z, using a weighted sum operation. T_i indicates the number of hidden states of the i-th feature. The fused representation is computed by

$$z = \frac{\sum_{i=1}^n w_i z_{T_i}^i}{\sum_{i=1}^n w_i};$$ (1)

where w_i is a trainable scalar weight and n is the number of feature types.

We initialize the first hidden state $h_0 = z$ in the decoder and use a GRU to generate the answer. The decoding procedure follows $h_{t+1} = \text{GRU}(y_t; h_t)$, where h_t is the t-th hidden state in the decoder and y_t is the output in the t-th step.

$$\Pr(y_t \mid y_{1:t-1}) = \text{softmax}(W h_t + b);$$ (2)

where $W \in \mathbb{R}^{j \times d_h}$ and $b \in \mathbb{R}^j$ are learned weights. and d_h are the vocabulary and the dimension of the hidden state respectively.

Multi-Stage Fusion

In addition to the above basic fusion model, here we introduce a novel feature fusion mechanism using multi-stage attention. The attention enables the model to focus on a targeted area within a context that is relevant to answer the question (Antol et al. 2015; Xiong, Merity, and Socher 2016). With the attention mechanism, we can enhance or modify vectors set with the information from other vectors; it is called the fusion process where information of some vectors is fused into other vectors (Huang et al. 2018).

An general attention function can be described as computing the weighted sum of values, where the weight is given by the attention score between a query and a key via a
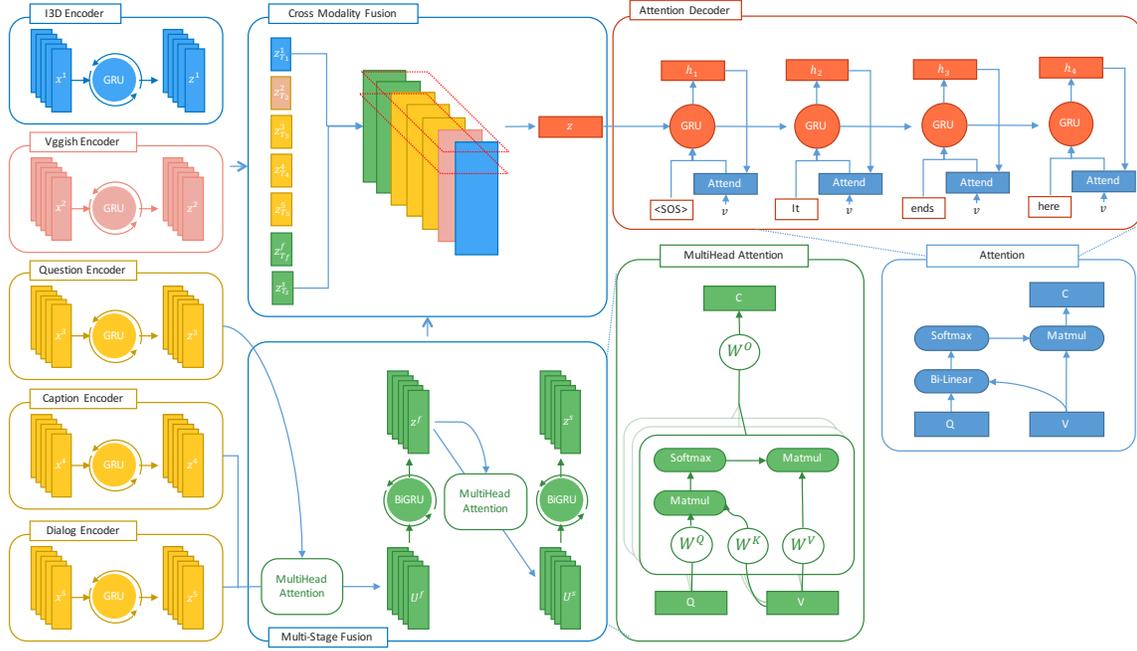
Figure 2: The illustraion of the proposed model architecture.

score function. For simplicity, our model uses the same vector for the value and the key. Here we choose the recently popular *multi-head attention* as our attention score function (Vaswani et al. 2017):

$$\text{MultiHead}(Q, V) = [\text{head}_1; \dots; \text{head}_n]W^O,$$
$$\text{head}_i = \text{Att}(QW_i^Q, VW_i^K, VW_i^V),$$
$$\text{Att}(Q, V, K) = \text{softmax}(\frac{QV^T}{\sqrt{d_k}})K,$$

(3)

where $W_i^Q \in \mathbb{R}^{d \times d_k}$, $W_i^K \in \mathbb{R}^{d \times d_k}$, $W_i^V \in \mathbb{R}^{d \times d_k}$ and $W^O \in \mathbb{R}^{nd_k \times d}$ are parameters. $[A; B]$ means the concatenation of two vectors A and B. n is the number of heads and d is the dimension. $d_k = d/n$. Note that we add a residual connection after the attention to help model converge faster (He et al. 2016).

Our multi-stage fusion starts from fusing encoded **question** (z^Q) into encoded **caption** (z^C) and **dialogue** (z^D), and we consider the concatenation of **caption** and **dialogue** to be the context. We apply the multi-head attention as (3) to generate fusion vectors and then feed them into a bidirectional GRU reader to obtain the fused representation:

$$U^{fusion} = \text{MultiHead}([z^C; z^D], z^Q)$$
$$z^{fusion} = \text{BiGRU}(U^{fusion})$$

(4)

While the context is necessary to infer the answer, an issue in our current fusion representation is lack of full understanding about the context. To address this problem, we compute the attention U^{self} using *self-attention*, which usually improves performance a lot in machine comprehension and question answering (Wang et al. 2017; Vaswani et al. 2017). We also feed U^{self} into a bidirectional GRU and obtain z^{self}, where z^{self} can be viewed as vectors that fuse all

textual information together. Formally written as:

$$U^{self} = \text{MultiHead}(z^{fusion}, z^{fusion}),$$
$$z^{self} = \text{BiGRU}(U^{self}).$$

(5)

Then we have tried many different ways to leverage z^{fusion} and z^{self}, and find the most efficient way is to use the last hidden state of them and add them into computation of the weighted sum in (1). Therefore, the multi-stage fusion mechanism intuitively offers richer information to enhance the first hidden state passed to the decoder.

1x1 Convolution Fusion

Instead of a simple weighted sum in (1), we find that it is beneficial to insert 1x1 convolution (Szegedy et al. 2015) before the weighted sum to help each feature channel interact with each other. The weighted sum operation can also be considered as an single channel 1x1 convolution with output normalization.

To model the multimodal features using n channels, we utilize the last hidden state for each feature type, $\{z_{T_1}^1, \dots, z_{T_n}^n\}$. We run n_c channels 1x1 convolution on the inputs and obtain $\{\hat{z}^1, \dots, \hat{z}^{n_c}\}$. Then we compute the weighted sum as (1):

$$z = \frac{\sum_{i}^{n_c} w_i \hat{z}^i}{\sum_{i}^{n_c} w_i},$$

(6)

where w_i is a trainable scalar weight.

Attention Decoder

In the decoding phase, attention is also commonly used to help model focus on important information in the encoder-decoder model (Luong, Pham, and Manning 2015). We

choose a commonly used multiplicative attention (Britz et al. 2017) and the computation follows:

$$a_{i;t} = v_i^T U^T V h_t;$$
$$\alpha_t = \text{softmax}(\boldsymbol{a}_t); \qquad (7)$$

where $U \in \mathbb{R}^{k \times d_x}$, $V \in \mathbb{R}^{k \times d_y}$ are trainable weights and $v_i \in \mathbb{R}^{d_x \times 1}$, $h_t \in \mathbb{R}^{d_y \times 1}$ are inputs. k is the attention dimension. Note that $a_{i;t} = v_i^T U^T V h_t$ is also called low-rank bilinear method (Pirsiavash, Ramanan, and Fowlkes 2009; Kim et al. 2016).

In the decoding step t, the attention of a query h_t and values $\boldsymbol{v} = \{v_1; \ldots; v_r\}$ is computed as:

$$C_t = \sum_i \alpha_{i;t} v_i: \qquad (8)$$

We then concatenate the context vector C_t with the next step input y_t as the attention-enhanced input and pass it through GRU, formally written as:

$$h_{t+1} = \text{GRU}([y_t; C_t]; h_t): \qquad (9)$$

Here we choose a simple concatenation of all encoded features to form $\boldsymbol{v}$: $\{z_1^1; \ldots; z_{T_1}^1; z_1^2; \ldots; z_{T_n}^n\}$. We will discuss how $\boldsymbol{v}$ influences performance in the experiments.

Top-Down Attention LSTM

We also try an alternative attention decoder called *top-down attention LSTM* (Anderson et al. 2018), which has been proved useful in visual question answering. The original design makes this model selectively attend to spatial image features, but we generalize it and feed arbitrary encoded values set $\boldsymbol{v} = \{v_1; \ldots; v_r\} = \{z_1^1; \ldots; z_{T_1}^1; z_1^2; \ldots; z_{T_n}^n\}$ into this model. We consider this model as an enhanced attention decoder, where an additional attention LSTM is used to further track what has been attended and expect it can provide more useful information for the attention module.

Here the top-down attention LSTM has two LSTM layers, which are named *attention LSTM* and *language LSTM*. We consider the first LSTM layer as an attention model and the second LSTM layer as a language model. The main purpose of separating two LSTM cells is to modularize the learning process into two parts: capturing the importance of multimodal features (attention LSTM) and language generation (language LSTM).

The input vector to the attention LSTM at time step t consists of the previous output hidden states of the language LSTM concatenated with the mean pooled features $\bar{v} = \frac{1}{r} \sum_{i=1}^r v_i$ and a word embedding of the previously generated word. $\boldsymbol{x}_t^1 = [h_{t-1}^2; \bar{v}; \text{embed}(y_{t-1})]$, where $\text{embed}(y_{t-1})$ is the word embedding of y_{t-1}. Given the output h_t^1 of the attention LSTM, we compute the attention score as follows:

$$b_{i;t} = w_a^T \tanh(W_{va} v_i + W_{ha} h_t^1);$$
$$\beta_t = \text{softmax}(\boldsymbol{b}_t): \qquad (10)$$

Then we calculate the context vector of attended features as

$$C_t = \sum_i \beta_{i;t} v_i: \qquad (11)$$

The input to the language LSTM consists of the context vector of attended vectors concatenated with the output of the attention LSTM, $\boldsymbol{x}_t^2 = [h_t^1; C_t]$. We pass $\boldsymbol{x}_t^2$ into the language LSTM to obain h_t^2 and compute the output word distribution as:

$$\Pr(y_t \mid y_{1:t-1}) = \text{softmax}(W h_t^2 + b); \qquad (12)$$

which is exactly same as the basic decoding setup in (2).

Training and Testing

The full model is trained in an end-to-end manner to optimize the dialogue generation results in (2). During testing, beam search is applied to generate the fluent answering sentences given the contexts.

Experiments

To evaluate whether the proposed model is capable of modeling dialogues with multimodal scenarios, a set of experiments is conducted.

Experimental Setup

We run our experiments using the official AVSD dataset (Alamri et al. 2018a), which consists of 7659, 1787, 1710 dialogues for train, dev, and test sets respectively. In the dataset, almost all dialogues contain 10 question/answer pairs. We use the evaluation script `nlg-eval` [1] to compute objective evaluation scores for the model, where the punctuations, ',' and '.', are not taken into account. In the experiments, the compared baselines include a naïve copy baseline and the released baseline (Alamri et al. 2018b), where the naïve baseline simply copies the questions as its answer and serves as a lower bound of this task. Note that because we found there are many word overlaps between the question and the answer, the copy baseline in fact has very strong performance under objective evaluation metrics.

Fine-Grained Response Revision

In the answering sentence to the yes/no question, adding "yes" or "no" at the beginning may affect the performance. Therefore, we train a classifier to predict whether "yes" or "no" tokens should be inserted. This classifier consists of an RNN encoder followed by a linear projection output layer. Note that we first filter out yes/no questions by rules, since yes/no questions should not be answered with this token.

Results

Table 2 shows the experimental results, where our models only take **vggish**, **caption**, and **dialog** into account, while the released baseline additionally use **i3d** as their input features. While the basic model is the simple architecture without any mechanism, it shows surprising performance specific in CIDEr (from 0.69 to 0.99) (Vedantam, Zitnick, and Parikh 2015). Because CIDEr measures the ability of capturing correct objects in the image and places less attention on words which frequently appear in answers such as

---
[1]https://github.com/Maluuba/nlg-eval

	Model Encoder	Model Decoder	B-1	B-2	B-3	B-4	MET.	R-L	CIDEr
Baseline	Naïve Copy		0.231	0.124	0.077	0.049	0.111	0.235	0.637
	Released (Alamri et al. 2018b)		0.270	0.172	0.118	0.085	0.115	0.292	0.790
Basic Fusion	Simple Fusion	Simple	0.232	0.157	0.112	0.084	0.120	0.305	0.994
	Multi-Stage	Simple	0.231	0.157	0.113	0.086	0.119	0.308	1.009
	1x1 Convolution	Simple	0.239	0.162	0.117	0.088	0.122	0.310	1.013
	Simple Fusion	Attention	**0.245**	0.162	0.115	0.086	0.119	0.308	0.977
	Simple Fusion	Top-Down Attention LSTM	0.234	0.158	0.113	0.085	0.119	0.309	0.986
Multi-Stage	Multi-Stage	Attention	0.238	0.161	0.116	0.088	0.123	0.311	1.028
	Multi-Stage + 1x1 Conv	Attention	0.238	0.163	0.118	0.090	0.122	**0.315**	**1.059**
	Multi-Stage + 1x1 Conv	Attention (w/o **vggish**)	0.243	**0.165**	**0.119**	**0.091**	**0.124**	0.313	1.046
	Multi-Stage + 1x1 Conv	Attention (w/ **i3d**)	0.234	0.157	0.112	0.084	0.117	0.303	0.958
Submitted	Basic Fusion Model (Prototype)		0.237	0.161	0.116	0.088	0.121	0.310	1.015
	+ Fine-Grained Response Revision		0.238	0.161	0.116	0.087	0.122	0.315	1.024
	Fusion Text Model		0.214	0.140	0.098	0.073	0.110	0.286	0.859
	+ Fine-Grained Response Revision		0.215	0.141	0.099	0.073	0.111	0.291	0.874

Table 2: The results of baselines and our proposed models, where our models do not use **i3d** and **summary**.

stopwords, the result tells that the basic model already captures important information and objects in the video without the complicated design of attention. In the proposed model, we analyze the results for different encoding and decoding mechanisms.

Encoding   We examine the effectiveness of the proposed two fusion methods, multi-stage fusion and 1x1 convolution fusion. The proposed multi-stage fusion provides model deeper understanding of input features and then further boosts the performance of CIDEr. The 1x1 convolution fusion also enable interactions between different modalities and improves the performance.

Decoding   Furthermore, two decoding mechanisms, attention decoder and top-down attention LSTM, are investigated. Two variants of decoders have different advantages. The attention decoder mainly improves BLEU scores (Papineni et al. 2002), while the top-down attention boosts the performance for the ROUGE score (Lin 2004). It may be because the attention decoder enhances the ability of our model to capture the specific word usage in dialogues. On the other hand, the top-down attention LSTM adds the attention LSTM to help track what is attended in previous decoding, and thus improves the ROUGE-L where the recall of generated words is relatively important. Hence, the attention decoder is applied as our decoder in the following experiments due to its strong ability of learning how to use particular word in certain context and think these modules will have complementary effects.

Proposed Fusion Model   Our full fusion model is composed of multi-stage fusion, 1x1 convolution fusion, and the attention decoder, and further improvement is investigated and analyzed in the experiments shown in Table 2. Compared to the released baseline, we can find that even the proposed basic model can outperform it by a large margin in CIDEr, improve METEOR and ROUGE-L, and

Method	B-1	B-2	B-3	B-4	MET.	R-L	CIDEr
Prod	.221	.146	.103	.076	.109	.288	.823
Sum	.232	.156	.111	.083	.116	.305	.950
Concat	**.237**	**.160**	**.116**	**.087**	**.120**	**.306**	.987
Weight	.232	.157	.112	.084	**.120**	.305	**.994**

Table 3: Results of different fusion methods.

achieve comparable results in BLEU-4. Note that BLEU-1 and BLEU-2 are not good metrics to evaluate quality of generated sentences in dialogues, because they only measure unigram and bigram precision. The copy baseline shows that we can reach high BLEU-1 and BLEU-2 by simply copying the questions, implying that these are not good indicators. The comparable BLEU-4 shows that our basic model in fact have similar ability to capture the sentence structure compared to the released baseline. Furthermore, our proposed full fusion model can significantly outperforms the released baseline for almost all metrics.

Here we also conduct experiments on two variants of our full fusion model. The first one does not use **vggish** features, so it is a pure text model, and the second one additionally uses **i3d** features, which can be fairly compared with the released baseline. Table 2 shows that the pure text model also achieves better BLEU scores compared to the model that uses additional video features, implying that the language part of test data provides rich enough information for the model to answer most questions. On the other hand, using **i3d** features results in performance drop, confirming our concern about noisy **i3d** features and demonstrating the need of the proposed feature selection.

Analyzing Fusion Operations

In Table 3, we test different representation fusion methods to fuse features $\{z^1_{T_1}, \ldots, z^n_{T_n}\}$ into v, while we choose weighted sum in the simple model (Table 2). 3 different fusion methods are listed below:
- Product: $z = \prod_i^n z^i_{T_i}$

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr	Human
Baseline (i3d) (Alamri et al. 2018b)	.621	.48	.379	.305	.217	.481	.733	—
Baseline (i3d + vggish) (Alamri et al. 2018b)	.626	.485	.383	.309	.215	.487	.746	2.848
Basic Fusion Model (Official)	.636	.510	**.417**	**.345**	**.224**	**.505**	**.877**	—
Basic Fusion Model (Prototype)	.640	**.513**	.416	.342	.223	.504	.837	**3.188**
+ Fine-Grained Response Revision	**.641**	**.513**	.416	.342	.223	.504	.836	—
Fusion Text Model	.592	.468	.375	.304	.206	.475	.729	2.928
+ Fine-Grained Response Revision	.595	.477	.376	.304	.207	.477	.731	—

Table 4: The submitted results in the official test set.

Features	B-1	B-2	B-3	B-4	MET.	R-L	CIDEr
dialogue	.228	.153	.110	.083	.116	.296	0.934
+ caption	.229	.154	.111	.084	.119	.305	0.976
+ vggish	.232	.157	.112	.084	.120	.305	0.994
+ i3d	**.239**	**.161**	**.116**	**.087**	**.121**	**.309**	**1.002**

Table 5: Results of the feature ablation tests. Model architectures are simple model using different input feature set.

Features	B-1	B-2	B-3	B-4	MET.	R-L	CIDEr
vggish	.233	.159	**.115**	**.087**	**.121**	**.309**	1.019
caption	.232	.158	.114	**.087**	.120	.307	1.007
dialogue	.234	.159	**.115**	**.087**	.120	**.309**	**1.023**
all	**.245**	**.162**	**.115**	.086	.119	.308	0.977

Table 6: Results of the attention ablation tests. Decoder is attention decoder which attends on different feature set of v.

- Sum: $z = \sum_i^n z_{T_i}^i$
- Concat: $z = W^{cat}[z_{T_1}^1; \ldots; z_{T_n}^n], W^{cat} \in \mathbb{R}^{d_h \cdot dn}$

As we expect, the weighted sum outperforms other methods except concat. While weighted sum is better in CIDEr, concat outperforms it in BLEU scores. Note that the number of parameters in the weighted sum is significantly smaller than the number in concat, because it only needs to learn n scalars and concat uses $d_h dn$ parameters. Considering the model size and extensibility, the weighted sum is considered as a better choice.

Ablation Test

In order to investigate the usefulness of features and attention, two sets of experiments are conducted for analysis.

Feature   In Table 5, we examine the effectiveness of each feature set by fixing the model architecture to the basic model. The results demonstrate that all feature sets contains semantically meaningful information, and the performance of the basic model gradually improves when we add more features. When using a simple model architecture, using **i3d** can improve the performance, which is different from the result shown in Table 2. The reason is probably that **i3d** is too noise to learn, and thus we observe these different results when using **i3d** features.

Attention   We also test our attention decoder on different attention values v. Different from our expectation, results

in Table 6 show that different attention values lead to little difference of performance. In sum, **dialogue** gives the best CIDEr score, and using all input features together leads to the best average BLEU score.

Official Results

We submit the following five predictions:
- Basic fusion model (official): a basic simple fusion model trained on official data
- Basic fusion model (prototype): a basic simple fusion model trained on prototype data (w/o and w/ fine-grained response revision)
- Fusion text model: a fusion model with multi-stage fusion and attention decoder that only take texts into consideration (w/o and w/ fine-grained response revision)

Note that the submitted models here used different experiment settings from the one proposed in this paper and the models were not well trained. Here feature selection is not applied, so the basic fusion model considers the **i3d** features. The results are shown in Table 4, and the performance is evaluated with consideration of multiple correct responses. Because the numbers in Table 4 are not comparable with ones in Table 2, we also show the submitted results in the last two rows of Table 2 for fair comparison. Our simple model outperforms the released baseline in all metrics, especially in CIDEr (by 12%) and human evaluation (by 11%). The proposed fine-grained response revision brings the small improvement, although the model which is not well trained.

Our proposed basic fusion architecture has less parameters but achieves better performance than the released baseline, which uses a complicated attention scheme. The results show that the simple 1×1 convolution fusion may have potential of disentangling each dimension on z shown in Figure 2 and then resulting in more semantically meaningful representations for multiple modalities.

Conclusion

This paper proposes an intuitive and effective visual dialogue model based on an encoder-decoder design. We develop a set of modules that are capable of fusing multimodal features with and performing context-aware decoding. In the experiments of the audio visual scene-aware dialogue task, we validate the effectiveness of each module and analyze whether it is necessary to use all features to correctly answer questions. The attempt in this paper bridges different modalities and encourages further model exploration for advancing the important research area about multimodality.

References

Alamri, H.; Cartillier, V.; Lopes, R. G.; Das, A.; Wang, J.; Essa, I.; Batra, D.; Parikh, D.; Cherian, A.; Marks, T. K.; et al. 2018a. Audio visual scene-aware dialog (avsd) challenge at dstc7. *arXiv preprint arXiv:1806.00525*.

Alamri, H.; Cartillier, V.; Lopes, R. G.; Das, A.; Wang, J.; Essa, I.; Batra, D.; Parikh, D.; Cherian, A.; Marks, T. K.; et al. 2018b. Audio visual scene-aware dialog (avsd) challenge at dstc7. *arXiv preprint arXiv:1806.00525*.

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, volume 3, 6.

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Lawrence Zitnick, C.; and Parikh, D. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, 2425–2433.

Britz, D.; Goldie, A.; Luong, T.; and Le, Q. 2017. Massive Exploration of Neural Machine Translation Architectures. *ArXiv e-prints*.

Carreira, J., and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, 4724–4733. IEEE.

Das, A.; Kottur, S.; Gupta, K.; Singh, A.; Yadav, D.; Moura, J. M.; Parikh, D.; and Batra, D. 2017. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hershey, S.; Chaudhuri, S.; Ellis, D. P.; Gemmeke, J. F.; Jansen, A.; Moore, R. C.; Plakal, M.; Platt, D.; Saurous, R. A.; Seybold, B.; et al. 2017. Cnn architectures for large-scale audio classification. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, 131–135. IEEE.

Huang, H.-Y.; Zhu, C.; Shen, Y.; and Chen, W. 2018. Fusionnet: Fusing via fully-aware attention with application to machine comprehension. In *International Conference on Learning Representations*.

Kim, J.-H.; On, K.-W.; Lim, W.; Kim, J.; Ha, J.-W.; and Zhang, B.-T. 2016. Hadamard product for low-rank bilinear pooling. *arXiv preprint arXiv:1610.04325*.

Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

Luong, T.; Pham, H.; and Manning, C. D. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, 311–318. Association for Computational Linguistics.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.

Pirsiavash, H.; Ramanan, D.; and Fowlkes, C. C. 2009. Bilinear classifiers for visual recognition. In *Advances in neural information processing systems*, 1482–1490.

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15:1929–1958.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, 3104–3112.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Vedantam, R.; Zitnick, C. L.; and Parikh, D. 2015. Cider: Consensus-based image description evaluation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 4566–4575.

Wang, W.; Yang, N.; Wei, F.; Chang, B.; and Zhou, M. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 189–198. Association for Computational Linguistics.

Xiong, C.; Merity, S.; and Socher, R. 2016. Dynamic memory networks for visual and textual question answering. In *International conference on machine learning*, 2397–2406.