The World in My Mind: Visual Dialog with Adversarial Multi-modal Feature Encoding

Yiqun Yao[123], **Jiaming Xu**[12*] **and Bo Xu**[1234]
[1]Institute of Automation, Chinese Academy of Sciences (CASIA). Beijing, China
[2]Research Center for Brain-inspired Intelligence, CASIA
[3]University of Chinese Academy of Sciences
[4]Center for Excellence in Brain Science and Intelligence Technology, CAS. China
{yaoyiqun2014,jiaming.xu,xubo}@ia.ac.cn

Abstract

Visual Dialog is a multi-modal task that requires a model to participate in a multi-turn human dialog grounded on an image, and generate correct, human-like responses. In this paper, we propose a novel Adversarial Multi-modal Feature Encoding (AMFE) framework for effective and robust auxiliary training of visual dialog systems. AMFE can force the language-encoding part of a model to generate hidden states in a distribution closely related to the distribution of real-world images, resulting in language features containing general knowledge from both modalities by nature, which can help generate both more correct and more general responses with reasonably low time cost. Experimental results show that AMFE can steadily bring performance gains to different models on different scales of data. Our method outperforms both the supervised learning baselines and other fine-tuning methods, achieving state-of-the-art results on most metrics of VisDial v0.5/v0.9 generative tasks.

1 Introduction

In recent years, there has been a rising attention in Artificial Intelligence on how to train a model to understand visual inputs from the physical world, and communicate them with human language. Typical problems include Visual Question Answering (VQA) (Antol et al., 2015) and Image Captioning (Xu et al., 2015). These tasks require a model to read an image and generate a proper response, such as answering a question grounded on the image, or generating a sentence to describe the image. As a more difficult extension, Visual Dialog (De Vries et al., 2017; Das et al., 2017a; Mostafazadeh et al., 2017) is a cluster of tasks featuring two agents conducting a multi-turn dialog grounded on an image. A model is usually trained to predict every single response of one agent in the two, based on the image and dialog history. There are also some different task settings such as directly training two agents to complete a goal-driven cooperative task such as Guessing Game (Das et al., 2017b).

Tasks involving both the physical world (visual images) and abstract world (languages) share a core issue: how to establish connections between these two worlds, and is there a framework to leverage these connections for learning? Temporarily, the majority of answers are learning end-to-end models with multi-modal feature fusion (Kim et al., 2016; Fukui et al., 2016; Yu et al., 2018). These methods usually merge the visual and language features into rich representations containing information from both sides. Some cross-modal attention methods (Lu et al., 2016; Nam et al., 2017) formulate the visual-language connections explicitly by parameterizing the attention weights to learn whether there is high correlation within certain pairs of language and visual feature vectors. However, in all these works, the merged representations or attention weights are only learned from pairwise (one image, one sentence) co-occurrence, and serve for the optimization of a loss function only related to the final ground-truth response. In fact, the features from both sides are not truly connected in an aspect of general distributions, but only merged into a new vector for each training/testing sample. We suppose that this is not good enough for a model to distill knowledge from both of the two worlds because the language/visual vectors do not contain knowledge from the other modality in the bottom level before they are merged.

In this paper, we discuss another possibility. We want to establish an unsupervised framework of multi-modal encoding, which directly generates an "image feature distribution" from a language

[*] Corresponding Author

Proceedings of NAACL-HLT 2019, pages 2588–2598
Minneapolis, Minnesota, June 2 - June 7, 2019.

distribution, or vice versa. For example, when a neural network based model receives a natural language sentence x as input, it encodes x into a sequence of high-dimensional continuous vectors. All these language vectors can be projected into another latent space to have a new distribution p_l. We train the language encoder to let the new distribution p_l be the same as, or very close to, the distribution p_v of all *image features* observed and encoded in the task data. Since we can partly recover a real-world image distribution from the language vectors achieved in this way, these language vectors intrinsically contain both language semantics and real-world image properties. This is a higher-level connection between the two worlds.

In order to train a model to generate samples subject to a certain distribution p_v from an original distribution p_l, Generative Adversarial Networks (GANs) have been proved very effective (Goodfellow et al., 2014; Arjovsky et al., 2017; Miyato et al., 2018). Lample et al. (2018) used adversarial training on the vectors produced by sentence encoders for different languages in unsupervised machine translation. However, different languages in their task are in single modality and share encoder structures, making the same method not directly usable and extendable for multi-modal tasks with largely different prior distributions and complex encoder structures with attention. In our work, we propose Adversarial Multi-modal Feature Encoding (AMFE), a novel GAN-based training schedule with an attention-based sample-selecting method, which can successfully force the multi-modal vectors to have closely related distributions, benefitting the performances of various visual dialog systems.

We test our method on the VisDial (Das et al., 2017a) benchmark (one example is shown in Table 1). A normal sample of VisDial contains an image and 10 turns of question-answering dialog from two people grounded on the image. A series of models have been proposed to solve the task, including memory and attention based models (Das et al., 2017a), reinforcement learning (Das et al., 2017b), knowledge transfer techniques (Lu et al., 2017) and GAN (Wu et al., 2017). Wu et al. (2017) designed a complex attention model and applied GAN in a traditional way to force the generated tokens to mimic real-world language (language vs. language), making their model only trainable through sequence sampling and reinforcement learning. Our work, on the other hand, applies a directly differentiable GAN on continuous vectors as a multi-modal feature encoding method (language vs. image).

	Caption: A dog with goggles is in a motorcycle side car
	A(1): can you tell what kind of dog this is
	B(1): he looks like beautiful pit bull mix
	A(2): can you tell if motorcycle is moving or still
	B(2): it's parked
	A(3): is dog's tongue lolling out
	B(3): not really

Table 1: An example from VisDial dataset.

Our contributions include:

- We propose AMFE: a novel Adversarial Multi-modal Feature Encoding framework to benefit visual dialog models. The core idea is to force features from different modalities to have closely related distributions.
- We develop efficient AMFE implementations, including a novel attention-based sample selecting method, for various commonly-used visual dialog models.
- Experimental results show that AMFE brings robust performance gains to different visual dialog models. We achieve state-of-the-arts on most metrics of VisDial v0.5/v0.9 generative tasks.

2 Related Work

2.1 Visual Dialog

Visual Dialog is a cluster of tasks sharing two properties: multi-turn and cross-modality. VisDial (Das et al., 2017a) is a widely-used benchmark with question-answering style dialogs grounded on real-world images. As a special case of dialog generation tasks, VisDial share some of the research concerns with single-modal natural language dialog generation (Dhingra et al., 2016; Serban et al., 2017; Sordoni et al., 2015; Serban et al., 2016; Liu et al., 2016). Natural language dialogs are usually discrete, state-dependent and style-free, thus some reinforcement learning (RL) methods have been proposed (Li et al., 2016). Das et al. (2017b) built an cooperative image guessing task on VisDial: they train both the questioner and the answerer, making them complete a same goal to help the questioner produce a guessing or "imagination" of the unseen image described by the answerer. The distance between the guessing and the target image is used as reward for reinforcement learning. In some extreme settings, such a task definition can even lead to emergence of a new language between robots (Kottur et al., 2017). After per-training, using their reinforcement learning method as an auxiliary loss can also bring performance gain in standard VisDial metrics such as mean rank.

However, generating a reward based on just one target image for a training sample may lead to a kind of overfitting. Language is highly abstract: one dialog can correctly describe a lot of different scenes in real world, so why should we force a dialog to fit one single example among them? Therefore, generating a reward from adversarial training is a more efficient way because it goes beyond individual samples into distributions. There are two previous works (Wu et al., 2017; Lu et al., 2017) that use GAN-like methods to boost the performances of pre-trained VisDial models. (Wu et al., 2017) proposes to use adversarial reinforcement learning. A discriminator is trained to distinguish the tokens of real/generated answers, and the answerer (generator) is trained via RL using a reward related to the score given by discriminator. This method is very effective, but using both RL Monte Carlo and GAN brings high computational cost. Also, a lot of tricks are involved for a good training. Our method, on the other hand, does not need Monte-Carlo sampling to compute immediate reward while generating each of the N words in a sentence ($O(N)$ time cost). (Lu et al., 2017) uses a knowledge-transferring method between generative and discriminative task settings. However, this requires the models on both settings to be pretrained well enough. Our work is also an adversarial learning based method, but it is more robust, time-efficient and effective.

2.2 GAN for Generative Tasks

GAN (Goodfellow et al., 2014; Arjovsky et al., 2017; Miyato et al., 2018) has raised much attention because of its ability to directly generate samples subject to a target distribution. Many training techniques have been proposed to solve the unstable training problems of GAN (Gulrajani et al., 2017; Kurach et al., 2018). Wasserstein GAN (WGAN) (Arjovsky et al., 2017) is a successful method using critic learning loss and weight clipping operations. We borrow some ideas from WGAN in the adversarial training of our model.

GAN well suits the image generation tasks because image signals are continuous and thus differentiable, enabling the gradient directly flowing back from the discriminator to generator. In language generation tasks, however, how to deal with the discrete sequence of symbols generated by the generator has long been a problem. A widely-used solution is applying RL with rewards generated by the discriminator (Wang et al., 2018; Li et al., 2017). As mentioned above, this is time-costing because RL needs to explore a large action space by sampling multiple action sequence. Besides, how the immediate reward is computed after generating each word is also a difficult problem.

Another solution is to avoid the discrete problem by applying adversarial training on the hidden states of the generator. This requires that there is a known distribution p for the hidden states we want the model to generate. A successful case is reported by (Lample et al., 2018): using adversarial training to restrict the hidden states of source language and target language (both from vanilla LSTMs) into a same latent space can boost the performance of unsupervised machine translation. Our AMFE framework is also an adversarial training on the language hidden states, but we are the first to use this kind of methods to establish connections between different modalities. Our training procedure is also largely different from (Lample et al., 2018) with our modified WGAN-like algorithm and a novel attention-based sample selection method: they are critical for training convergency on multi-modal tasks, with complex attention-based model structures.

3 Model

We first define the task and our framework formally, and then describe how it is implemented and trained on different visual dialog models.

3.1 VisDial Task Definition

In the VisDial task, each sample contains an image I, a caption sentence C and a dialog D with $T = 10$ turns in total. In each turn t, there is a question q_t about the image, and a ground truth answer a_t. The model needs to read the dialog history $H = \{C, (q_1, a_1), ..., (q_{t-1}, a_{t-1})\}$ and image I, to generate an answer as a response to q_t. We rewrite $H_t = (q_t, a_t)$ and $H_0 = C$. Formally, the dialog agent (named A-Bot) outputs an answer

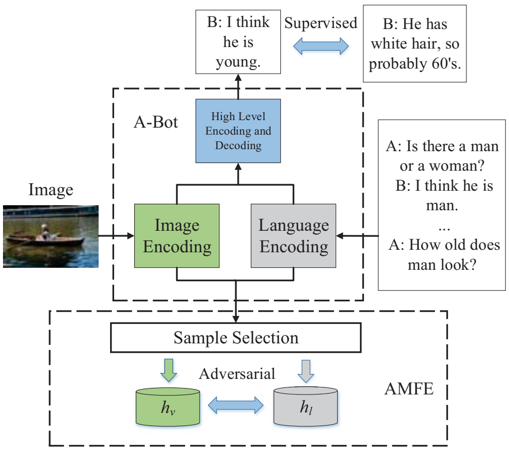


Figure 1: AMFE framework with a generative encoder-decoder model and multi-modal adversarial training.

prediction $\hat{a}_t$:

$$\hat{a}_t = ABot(q_t, I, H_{0\sim t-1}). \tag{1}$$

3.2 AMFE Framework

The goal of our Adversarial Multi-modal Feature Encoding (AMFE) is to restrict the distribution of feature representations from one modality m_1 to be closely related to that from another modality m_2. We take $m_1 = l(anguange)$ and $m_2 = v(isual)$. Specifically, A-Bot encodes language inputs into vectors h_l, and visual inputs into h_v, respectively. We want h_l and h_v to have indistinguishable distributions:

$$h_l, h_v \sim p(h). \tag{2}$$

To achieve this goal, we use a discriminative model (named D-Bot) to classify whether a vector encoded by A-Bot comes from modality l or v. D-Bot is trained with real h_l and h_v samples, while A-Bot is trained to generate language vectors h_l that can confuse D-Bot to classify them as label v. Figure 1 shows our framework.

3.3 A-Bot

We implement our AMFE method on two commonly-used visual dialog models, using them as A-Bot. The two A-Bot models are named Hierarchical Recurrent Encoder (HRE) and History-Conditioned Image Attentive Encoder (HCIAE), respectively. A-Bot learns to predict the right answer in each turn. In this process, it also encodes language and visual inputs into h_l and h_v samples, which we use for AMFE training.

3.3.1 Hierarchical Recurrent Encoder (HRE)

HRE is a hierarchical LSTM (Hochreiter and Schmidhuber, 1997) model used in (Das et al., 2017a,b). In HRE, a pre-trained Convolutional Neural Network (CNN) encodes the image into a single feature vector, which is further mapped into a visual representation I by a trainable Multi-Layer Perceptron (MLP). In each turn t, the question is encoded by a word-level LSTM into a question vector q_t, and the dialog history in the previous step H_{t-1} is encoded by another LSTM into vector f_{t-1}. There is a state-tracker $LSTM^{st}$ on the top level: $LSTM^{st}$ is forwarded one step each turn, integrating all the encoded vectors mentioned above. It reads the encoded history f_{t-1}, image vector I, the question q_t and the previous hidden state s_{t-1} from itself, and produces the new hidden state representation s_t:

$$s_t = LSTM^{st}([q_t, I, f_{t-1}], s_{t-1}), \tag{3}$$

where $[\cdot]$ stands for concatenation.

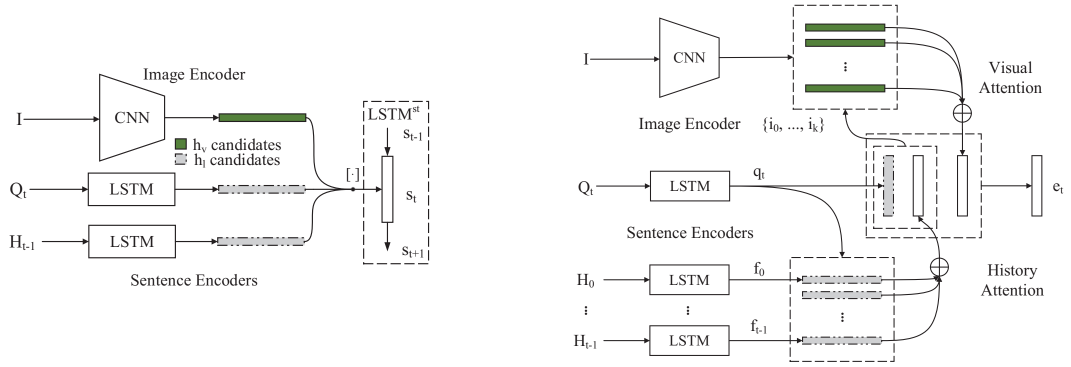
The answer decoder in HRE is an LSTM that takes s_t as initial state, and predicts one word at a time by a softmax probability over the vocabulary, to generate the whole answer sentence. Figure 2(a) shows the encoder structure of HRE. We use image vectors I as h_v samples (dark green) in AMFE, and both the q and f vectors as h_l samples (pink).

3.3.2 History-Conditioned Image Attentive Encoder (HCIAE)

HCIAE model (Lu et al., 2017) contains an textual attention on all history vectors based on the question, and a visual attention based on both the history and the question. In detail, it uses a pre-trained CNN to encode the image into a set of visual feature vectors V. Each vector in V is further passed through a trainable MLP, resulting in a visual feature set $\{i_0, ...i_{K-1}\}$. In each turn t, the question is encoded by an LSTM into vector q_t; the dialog history $\{H_0, H_1, ..., H_{t-1}\}$ is encoded by another word-level LSTM into vectors $\{f_0, ..., f_{t-1}\}$. The attention weight between q_t and each history vector f_j is computed as:

$$\begin{aligned} z_t^j &= \omega_a^T tanh(W_f f_j + W_q q_t), \\ \alpha_t^j &= softmax(z_t^j), \end{aligned} \tag{4}$$

where $\omega_a^T \in \mathbb{R}^{d\times 1}$, $W_f \in \mathbb{R}^{d\times d}$, $W_q \in \mathbb{R}^{d\times d}$ are trainable parameters; d is the length of both



(a) HRE encoder model.

(b) HCIAE encoder model.

Figure 2: A-Bot models we use for AMFE training. The intermediate vectors used as h_v and h_l candidates in AMFE are colored dark green and pink, respectively.

question and history features. A memory vector $\hat{m}_t$ is computed by:

$$\hat{m}_t = \sum_{j=0}^{t-1} \alpha_t^j f_j. \tag{5}$$

The memory vector is further used as a key to compute a similar visual attention over $\{i_0, ...i_{k-1}\}$ to achieve a final image vector $\hat{v}_t$. The final output of the encoder is computed by:

$$e_t = tanh(W_e[q_t, \hat{m}_t, \hat{v}_t]), \tag{6}$$

where $W_e \in \mathbb{R}^{d\times 3d}$ is trainable parameters; $[\cdot]$ stands for concatenation.

The answer decoder is an LSTM like that of HRE, taking e_t as input. Figure 2(b) shows the structure of HCIAE encoder. HCIAE produces more visual vectors for each image than HRE. We take all the q and f vectors as h_l candidates of AMFE, and the spatial visual features $\{i_0, ...i_{k-1}\}$ as candidates for h_v.

3.4 D-Bot

Despite the multiple choices of A-Bot, our D-Bot is always an MLP with two hidden layers of size 512 and ReLU activation. It is used to compute a loss function that forces all the h_l samples to be subject to the same distribution $p(h)$ as the visual vectors h_v. D-Bot takes a vector h in size d as its input, and predicts the probability of h coming from real image distribution and the visual encoder:

$$\hat{p}_v(modality = v|h) = DBot(h). \tag{7}$$

D-Bot is the discriminator from a GAN viewpoint. A-Bot must learn to confuse D-Bot in order to generate language features indistinguishable from image features.

3.5 Training

3.5.1 Loss Functions

To train our model, we use standard supervised training with cross-entropy loss function for pretraining, and add in adversarial training to produce an auxiliary loss to improve feature encoding.

The supervised learning loss is:

$$L_{su} = \frac{1}{N}\sum_{n=1}^{N} - \log(p(w_n^t|w_{<n}^t)), \tag{8}$$

where N is the full length of the decoded sentence.

For adversarial learning, A-Bot is trained to minimize the probability that D-Bot predicts the generated features to be fake samples. Following WGAN (Arjovsky et al., 2017), we do not use logarithm but directly optimize the likelihood itself:

$$L_{adv} = -E_{h_l}[DBot(h_l)]. \tag{9}$$

We sum L_{adv} as an auxiliary loss with a tunable weight λ, making A-Bot minimize:

$$L_G = L_{su} + \lambda L_{adv}. \tag{10}$$

On the other hand, D-Bot maximizes the following objective to distinguish real-world image vectors h_v from the language vectors h_l:

$$L_D = E_{h_v}[DBot(h_v)] - E_{h_l}[DBot(h_l)]. \tag{11}$$

We switch between A-Bot and D-Bot updates for each batch of dialog samples.

3.5.2 Attention-based Sample Selection

We have specified where the h_l and h_v samples come from while using different A-Bots in Section 3.3. Typically, for each batch of samples with batch-size M, in each turn t, there are M question vectors and $M \times t$ history vectors as h_l candidates. For HRE encoder, there are M image vectors as h_v candidates, while the number is $M \times K$ for the HCIAE encoder; K is the number of "pixels" in the final CNN feature-map. Thus, it is impossible to use all the generated samples in AMFE. For a successful training, the selected samples must be efficient, informative and balanced.

While using HCIAE, in order to compute L_{adv}, we use M question vectors and $M * w$ history vectors as h_l samples. The history vectors are selected using textual attention weights α_t^j produced by the temporary model: for each dialog, we pick the top w history vectors with the highest attention weights. We call this Attention-based Sample Selection (AbS). While computing L_D to train D-Bot, we use the same technique on the image, using the top-attended $M * w$ image vectors, together with another M image vectors randomly sampled from the dataset as positive samples h_v. The M question vectors and $M * w$ history vectors are used as a pool of negative samples h_l. In our experiments, $w = 1, 2$ works well.

While using HRE, since the model always "attends" on f_{t-1} by default (Eq. 3), we directly select q_t and f_{t-1} as h_l samples. We use the M image vectors I in this batch, together with another M image vectors randomly sampled from the dataset as the pool of h_v. The full training procedure is specified in Algorithm 1.

4 Experiments

4.1 The VisDial Dataset

VisDial is a visual dialog dataset based on MS COCO (Lin et al., 2014) images. There are 10 turns of human-posed question-answering dialogs on each image, with the questioner kept not seeing the image during the data collection process. For generative models, a model must give the probability of generating each candidate answer without seeing other candidates, and the rank of the ground-truth answer in the 100 candidates is used to compute different evaluation metrics; for discriminative models, the model can read and encode all the candidate answers and directly assign scores on them. According to the nature of GANs and similarities to real-world application scenarios, we use the generative setting for our model: it is equipped with a sequential decoder instead of a scoring module.

Algorithm 1 AMFE Training Procedure.

```
Require: $\alpha$ the learning rate; $c$ the clipping parameter; $M$ the batch size; $w_0$ the initial D-Bot parameters; $\theta_0$ the initial A-Bot parameters; dialog samples.
  Pre-train A-Bot with $L_{su}$ (Eq. 8).
  while $\theta$ has not converged do
      Sample $M$ turns of $(q, H, I, a)$ dialog samples.
      Forward A-Bot and select $h_l$ by attention weights.
      Compute $L_{adv} = -\sum_{k=1}^{M} DBot(h_l^k))$.
      Compute $L_{su}$ with ground-truth answers using (Eq. 8).

      Update $\theta$ to minimize $L_G = L_{su} + \lambda L_{adv}$.
      Switch to D-Bot training.
      Select image vectors $h_v$ by attention weights.
      Re-generate $h_l$ samples using the updated A-Bot.
      Compute $L_D = \sum_{k=1}^{M} DBot(h_v^k) - \sum_{k=1}^{M} DBot(h_l^k)$.
      Update $w$ to maximize $L_D$.
      Clip D-Bot weight $w$ into range $(-c, c)$.
  end while
```

For fair and sufficient comparison, we evaluate our model on both VisDial v0.5 and VisDial v0.9. VisDial v0.5 has 68k COCO images, for a total of 680k QA-pairs. Following (Das et al., 2017a) and (Das et al., 2017b), we use 50,729 images for training, 7,663 for validation and 9,628 for testing. Visdial v0.9 has 123,287 images. There are different splitting of train/valid/test in previous work. We follow (Lu et al., 2017) to use 82k for training, 1k for validation and 40k for testing. [1]

We compare our results to several existing models on the VisDial dataset, including:

- *Answer Prior* (Das et al., 2017a): directly encoding answer candidates with an LSTM and scoring by a linear model that captures the frequency of answers in the training set.
- *NN-QI* (Das et al., 2017a): a k-Nearest Neighborhood method considering only the question and the image. Unlike generative methods, both Answer Prior and NN-QI need to know the answer candidates.
- *LF-QIH-G* (Das et al., 2017a): a Late Fusion encoder that encodes the question, image and history separately. The encoded features are concatenated and linearly transformed to a

[1] VisDial has released v1.0 recently, and claims that models trained on v0.9 should also use the new v1.0 test set. Due to lack of baselines in the *generative* task, we follow the original widely-used settings of v0.5 and v0.9.

joint representation. The answer is produced by a generative decoder.

- *HRE* (Das et al., 2017b): the HRE model introduced in Section 3.3.
- *HREA-QIH-G* (Das et al., 2017a): a modified HRE A-Bot with attention to dialog history.
- *MN-QIH-G* (Das et al., 2017a): a Memory Network encoder that stores each piece of dialog history embeddings in an explicit memory. These embeddings can be attended and fused while generating the answer.
- *HCIAE* (Lu et al., 2017): the HCIAE model introduced in Section 3.3.
- *CoAtt* (Wu et al., 2017): this is a previous state-of-the-art model with a more complex co-attention encoder; the decoder is enhanced by adversarial reinforcement learning for better answer generation.

4.2 AMFE for HRE

We first test the efficiency of AMFE on the simpler A-Bot model: HRE. We use VisDial v0.5 as our benchmark for fair comparison with other HRE-based models and auxiliary training methods.

4.2.1 Implementation Details

For Visdial v0.5 dataset, we follow the preprocessing procedure and hyper-parameters described in (Das et al., 2017b). We pass each image through a pre-trained VGG-16 (Simonyan and Zisserman, 2015) CNN, and pick the single f7 vector as input image feature. We limit the maximum lengths of captions, questions and answers to be 40, 20 and 20, respectively; we remove words appearing less than 5 times in the training set, and replace them by a UNK token. We use vector size 300 for word embedding and 512 for all language and visual feature vectors. All LSTMs have two layers.

We pre-train A-Bot with L_{su} for 20 epochs before L_{adv} is added in. The batch-size is set to be 32. After each update of A-Bot, we perform 5 D-Bot updates. We use the 32 encoded image vectors in the batch, together with 32 image vectors randomly sampled from the dataset, to form 64 positive samples; for negative samples, we use the 32 question vectors and 32 history vectors $(t-1)$ from the updated A-Bot. We use Adam (Kingma and Ba, 2014) for A-Bot and RMSprop (Tieleman and Hinton, 2012) algorithm for D-Bot to perform gradient descending. The learning rate is set to 1e-3 for pre-training, further decayed to 5e-5; after adversarial training starts, the learning rate is fixed to 5e-5 for both A- and D-Bot. In the weight clipping step of WGAN (Arjovsky et al., 2017), we use a clipping parameter $c = 0.01$.

Model	MRR	R@1	R@5	R@10	Mean
Answer Prior	0.311	19.85	39.14	44.28	31.56
NN-QI	0.385	29.71	46.57	49.86	30.90
LF-QIH-G	0.430	33.27	51.96	58.09	23.04
HREA-QIH-G	0.442	34.47	53.43	59.73	21.83
MN-QIH-G	0.443	**34.62**	53.74	60.18	21.69
HRE-MLE	0.436	33.02	53.41	60.09	21.83
Frozen-Q-Multi	0.437	33.22	53.67	60.48	21.13
HRE-AMFE	**0.445**	**34.62**	**53.95**	**60.76**	**20.98**

Table 2: VisDial v0.5 evaluation results. The five metrics are mean reciprocal rank, recall of the ground-truth answer in the top-1/5/10 ranked candidates (higher is better), and the mean rank of the ground-truth answer (lower is better).

4.2.2 VisDial v0.5 Evaluation Results

On VisDial v0.5, two previous top models are a Memory Network based model (MN-QIH-G) by (Das et al., 2017a) and a multi-loss training on HRE encoder (Frozen-Q-Multi) based on goal-driven reinforcement learning (Das et al., 2017b). We start from the same HRE hyper-parameters and checkpoint as (Das et al., 2017b), but continue with our AMFE instead of reinforcement learning.

Table 2 shows the results on all the five evaluation metrics on VisDial v0.5. Results in the first 4 rows are copied from (Das et al., 2017a). AMFE achieves better performances than the supervised training of A-Bot model (HRE-MLE), especially significant on R@5, R@10 and mean rank, indicating that the adversarial feature encoding results in "generally better" dialogs. It also outperforms the another HRE-like model with history attentions (HREA-QIH-G). While used for multi-loss training, AMFE is significantly better than Frozen-Q-Multi, setting a new state-of-the-art on all metrics. We point out that in Frozen-Q-Multi (Das et al., 2017b), the goal-driven reinforcement leaning reward is computed pair-wise (considering how much can the questioner rebuild the image from the answerer's words), but the reward computed with a single image is not good enough to evaluate the dialog actions. This is because language is much more abstract than image, and failure to recover an image does not necessarily mean that the dialog is actually bad. Our method could avoid this issue because adversarial training is based on general distributions.

4.3 AMFE for HCIAE

In this section, we test the efficiency of AMFE for the HCIAE model with attention. We use VisDial v0.9 as our benchmark for fair comparison with (Lu et al., 2017).

4.3.1 Implementation Details

For Visdial v0.9 dataset, we follow the preprocessing procedure and HCIAE structure described in (Lu et al., 2017). We pass each image through a pre-trained VGG-19 CNN, resulting in a $512 \times 7 \times 7$ feature-map as visual input. To speed up convergence, we add a Batch Normalization (Ioffe and Szegedy, 2015) after the MLP that further encodes these visual vectors. We limit the maximum lengths of captions, questions and answers to be 24, 16 and 8, respectively. All LSTMs have only one layer.

HCIAE can be trained with either supervised loss (HCIAE-G-MLE) or with multi-loss involving knowledge-transfer (HCIAE-G-DIS). We test AMFE in both settings. For HCIAE-G-MLE, we pre-train HCIAE model with supervised loss for 20 epochs using learning rate 4e-4, and switch to AMFE training with learning rate 5e-5. For HCIAE-G-DIS, we start from the generative model trained with AMFE, together with a pre-trained HCIAE discriminative model. We follow the original knowledge-transfer training schedule, and add our L_{adv} to the original mixed loss function with weight 1. We use batch-size 32 for AMFE training, although the original paper used 128. Other settings are kept the same. For more details please see (Lu et al., 2017).

4.3.2 VisDial v0.9 Evaluation Results

Table 3 shows the results on v0.9. All the HCIAE results are picked from (Lu et al., 2017), and all CoAtt results are picked from (Wu et al., 2017); CoAtt-GAN-TF stands for training a CoAtt model with adversarial reinforcement learning and supervised teacher-forcing; HCIAE-AMFE stands for using AMFE on an HCIAE-G-MLE pre-trained model; HCIAE-GD-AMFE means using AMFE as an additional loss to join the HCIAE-G-DIS multi-loss training.

Model	MRR	R@1	R@5	R@10	Mean
Answer Prior	0.374	23.55	48.52	53.23	26.50
NN-QI	0.427	33.13	50.83	58.69	19.62
LF-QIH-G	0.520	41.83	61.78	67.59	17.07
HREA-QIH-G	0.524	42.28	62.33	68.17	16.79
MN-QIH-G	0.526	42.29	62.85	68.88	17.06
CoAtt-G-MLE	0.541	44.32	63.82	69.75	16.47
CoAtt-GAN-TF	**0.558**	**46.10**	65.69	71.74	14.43
HCIAE-G-MLE	0.539	44.06	63.55	69.24	16.01
HCIAE-G-DIS	0.546	44.35	65.28	71.55	14.23
HCIAE-AMFE	0.547	44.40	65.35	71.69	14.42
HCIAE-GD-AMFE	0.554	45.42	**66.09**	**72.30**	**14.11**

Table 3: VisDial v0.9 evaluation results. The five metrics are mean reciprocal rank, recall of the ground-truth answer in the top-1/5/10 ranked candidates, (higher is better) and the mean rank of the ground-truth answer (lower is better).

On VisDial v0.9, we observe that using AMFE on HCIAE can also boost the performances. Comparing HCIAE-G-MLE and HCIAE-AMFE, we can observe the same advantage over supervised training as on HRE, indicating that our method works for different dataset scales and A-Bot structures; comparing HCIAE-AMFE and HCIAE-G-DIS, AMFE is a competitive method for auxiliary training. Combining AMFE and HCIAE-G-DIS achieves better results than previous state-of-the-art (Wu et al., 2017) on R@5, R@10 and mean rank, and comparable on MRR and R@1. Besides, AMFE trains reasonably faster because we avoid the $O(N)$ time cost for Monte-Carlo sampling while computing temporary rewards (Wu et al., 2017).

We explain the efficiency of AMFE in two aspects. Firstly, AMFE is an adversarial training procedure forcing the language to be encoded into a distribution closely connected to the images. With attention-based sample selection, the most informative samples from both modalities are able to transfer knowledge. Secondly, like Batch Normalization, AMFE contributes to bring better numerical properties to the intermediate tensors in a network, especially on their means and variances, which could potentially benefit model performance.

4.4 Ablation Study

Both the weight of adversarial loss and the attention-based sample selection are critical to good performance. Table 4 shows ablation studies on these factors on HCIAE and VisDial v0.9.

Model	MRR	R@1	R@5	R@10	Mean
HCIAE-G-MLE	0.539	44.06	63.55	69.24	16.01
AMFE-standard	0.547	44.40	65.35	71.69	14.42
w/o AbS	0.537	44.03	63.52	69.33	15.45
$\lambda = 0.1$	0.538	44.10	63.59	69.18	16.07
$\lambda = 10$	0.534	44.01	62.83	68.64	16.56

Table 4: VisDial v0.9 ablation results training with HCIAE-G-MLE and AMFE fine-tuning. AMFE-standard uses $\lambda = 1$ with AbS selection.

#1 Caption: There are several airplanes parked on the tarmac.			
Question	Ground Truth	HRE-MLE	HRE-AMFE
1. are these commercial airplanes	yes they are	yes	yes
2. do they have company names on them	1 says UNK and other looks like it says UNK	yes , but i ca n't read them	yes , but i ca n't read them
3. are all of planes white	they are white 1 has blue and yellow on it and other 2 have red and blue	no they are white	no they are all white
4. are there people visible	yes there are 4 people visible	yes	yes
5. can you tell what sex they are	looks like they are all men	it is male	i think they are both male
6. are any of them wearing sunglasses	i think maybe 1 is it 's hard to tell they are kind of far away	i ca n't tell i can only see their backs	i ca n't tell , they are far away
7. are they dressed for warm or cold weather	looks like warm weather they are wearing their safety gear	they are wearing jackets and shorts	looks warm out
8. are there any trees visible	in far background there are lot of trees	no trees at all	yes there are trees in background
9. can you see any planes taking off or landing	no they all appear to be parked	no i do n't	no i do n't
10. are there any pilots visible inside	no i do not see any	no i do not see any	no there are not
#2 Caption: Some trucks move down a city road , while children play nearby.			
Question	Ground Truth	HRE-MLE	HRE-AMFE
1. how many children are there	i see 2	i can see 3 clearly 3 others are too far away	i see 1
2. girls or boys	bot boys	girls	i think they are boys
3. is there fire	no emergency	not that i can see	not that i can see
4. is anyone on fire truck that you can see	i can barely she driver	no it 's just him	no
5. what are kids playing	they look more like they are walking then playing	i think they are hockey	they are n't shown
6. does it look sunny out	not at all	yes	yes
7. is it black and white picture	no it is not	no	no
8. is it snowing	no it is not	no it is not	no it is not
9. is it raining	no , but it may soon	no it is not	no it is not
10. do you see dark clouds	yes i do	no	no

Table 5: Two examples in VisDial v0.5 dataset for case study.

4.5 Human Evaluation

The above results show that AMFE is especially strong at more "general" metrics such as R@5 and mean rank. To confirm that adversarial training on hidden states can help much to generate responses that are more natural, we randomly select 100 dialog samples from both VisDial v0.5 and v0.9 dataset, and ask two human subjects to vote for the responses generated by two groups of models: HRE-MLE vs. HRE-AMFE on v0.5, and HCIAE-G-MLE vs. HCIAE-AMFE on v0.9, both with beam-size 5. Model names are hidden while voting. We ask the human subjects to consider two metrics separately: (1) the fluency of the generated answer sentences and (2) the correctness of the answers compared to the ground-truths. As shown in Table 6, AMFE wins all the votes with different metric and different models, indicating that AMFE is robust in generating more natural responses.

Models	MLE Wins	AMFE Wins	Tie
HRE-fluency	30	52	18
HCIAE-fluency	34	43	23
HRE-correctness	33	42	25
HCIAE-correctness	29	38	33

Table 6: Human voting result on 100 samples from VisDial v0.5 and v0.9.

5 Case Studies

We randomly sample some dialogs from VisDial v0.5 validation set and illustrate the ground-truth answers and the generated answers with/without AMFE. Two results are shown in Table 5. In the first example, the model trained with AMFE generates a right vs. wrong answer in the 8-th turn, and a grammatically better response in the 5-th turn, compared to supervised pre-training. In the second example, the model trained with AMFE has a generally right understanding of the questions and the image, while the HRE-MLE model is generating response as if it does not see the image. This indicates that encoding language features in the image space leads to better understanding on both modalities.

6 Conclusion

We propose AMFE: an unsupervised multi-modal feature encoding framework and its implementations on different commonly-used visual dialog models. Our core idea is to force features from different modalities to have closely related distributions. Experiments show that AMFE can bring performance gains to both simple and complex models on different scales of VisDial dataset. Future work will possibly be visualizing the visual and language features encoded by AMFE to find more straightforward interpretations, as well as trying our method on more complex structures, discriminative models, and on discriminative tasks such as VQA and visual reasoning.

Acknowledgements

We thank the reviewers for their insightful comments. This work was supported by the National Natural Science Foundation of China (61602479), the Beijing Brain Science Project (Z181100001518006) and the Strategic Priority Research Program of the Chinese Academy of Sciences (XDB32070000).

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein gan. *arXiv preprint arXiv:1701.07875*.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017a. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2.

Abhishek Das, Satwik Kottur, José MF Moura, Stefan Lee, and Dhruv Batra. 2017b. Learning cooperative visual dialog agents with deep reinforcement learning. In *Proceedings of the IEEE international conference on computer vision*.

Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C Courville. 2017. Guesswhat?! visual object discovery through multi-modal dialogue. In *CVPR*, volume 1, page 3.

Bhuwan Dhingra, Lihong Li, Xiujun Li, Jianfeng Gao, Yun-Nung Chen, Faisal Ahmed, and Li Deng. 2016. Towards end-to-end reinforcement learning of dialogue agents for information access. *arXiv preprint arXiv:1609.00777*.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. Multimodal compact bilinear pooling for visual question answering and visual grounding. *arXiv preprint arXiv:1606.01847*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777.

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

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456.

Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. 2016. Hadamard product for low-rank bilinear pooling. *arXiv preprint arXiv:1610.04325*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Satwik Kottur, José MF Moura, Stefan Lee, and Dhruv Batra. 2017. Natural language does not emerge'naturally'in multi-agent dialog. *arXiv preprint arXiv:1706.08502*.

Karol Kurach, Mario Lucic, Xiaohua Zhai, Marcin Michalski, and Sylvain Gelly. 2018. The gan landscape: Losses, architectures, regularization, and normalization. *arXiv preprint arXiv:1807.04720*.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Unsupervised machine translation using monolingual corpora only. In *ICLR*.

Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. 2016. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

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 *Advances in Neural Information Processing Systems*, pages 314–324.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. 2018. Spectral normalization for generative adversarial networks. In *ICLR*.

Nasrin Mostafazadeh, Chris Brockett, Bill Dolan, Michel Galley, Jianfeng Gao, Georgios P Spithourakis, and Lucy Vanderwende. 2017. Image-grounded conversations: Multimodal context for natural question and response generation. *arXiv preprint arXiv:1701.08251*.

Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. 2017. Dual attention networks for multimodal reasoning and matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 299–307.

Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, volume 16, pages 3776–3784.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, pages 3295–3301.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *ICLR*.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

Tijmen Tieleman and Geoffrey Hinton. 2012. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31.

Xin Wang, Wenhu Chen, Yuan-Fang Wang, and William Yang Wang. 2018. No metrics are perfect: Adversarial reward learning for visual storytelling. *arXiv preprint arXiv:1804.09160*.

Qi Wu, Peng Wang, Chunhua Shen, Ian Reid, and Anton van den Hengel. 2017. Are you talking to me? reasoned visual dialog generation through adversarial learning. *arXiv preprint arXiv:1711.07613*.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.

Zhou Yu, Jun Yu, Chenchao Xiang, Jianping Fan, and Dacheng Tao. 2018. Beyond bilinear: Generalized multimodal factorized high-order pooling for visual question answering. *IEEE Transactions on Neural Networks and Learning Systems*, (99):1–13.