Boosting Naturalness of Language in Task-oriented Dialogues via Adversarial Training

Chenguang Zhu

Microsoft Speech and Dialogue Research Group, Redmond, WA, USA

`chezhu@microsoft.com`

Abstract

The natural language generation (NLG) module in a task-oriented dialogue system produces user-facing utterances conveying required information. Thus, it is critical for the generated response to be natural and fluent. We propose to integrate adversarial training to produce more human-like responses. The model uses Straight-Through Gumbel-Softmax estimator for gradient computation. We also propose a two-stage training scheme to boost performance. Empirical results show that the adversarial training can effectively improve the quality of language generation in both automatic and human evaluations. For example, in the RNN-LG Restaurant dataset, our model AdvNLG outperforms the previous state-of-the-art result by 3.6% in BLEU.

1 Introduction

In task-oriented dialogues, the computer system communicates with the user in the form of a conversation and accomplishes various tasks such as hotel booking, flight reservation and retailing. In this process, the system needs to accurately convert the desired information, a.k.a. *meaning representation*, to a natural utterance and convey it to the users (Table 1). The quality of response directly impacts the user's impression of the system. Thus, there are numerous previous studies in the area of natural language generation (NLG) for task-oriented dialogues, ranging from template-based models (Cheyer and Guzzoni, 2014; Langkilde and Knight, 1998) to corpus-based methods (Dušek and Jurčíček, 2016; Tran and Nguyen, 2017; Wen et al., 2015; Zhu et al., 2019).

However, one issue yet to be solved is that the system responses often lack the fluency and naturalness of human dialogs. In many cases, the system responses are not natural, violating inherent human language usage patterns. For instance,

Input	name[Wildwood], eatType[restaurant], food[Indian], area[riverside], familyFriendly[no], near[Raja Indian Cuisine]
with adv.	Wildwood is an Indian restaurant **in the riverside area near Raja Indian Cuisine**. It is not family friendly.
w/o adv.	Wildwood is a restaurant providing Indian food. It is **located in the riverside.** It is **near Raja Indian Cuisine**.

Table 1: Example of generated utterances from meaning representation input. Our model learns to put two pieces of location information in one sentence via adversarial training.

in the last row of Table 1, two pieces of location information for the same entity *restaurant* should not be stated in two separate sentences. In another example in Table 4, the positive review *child friendly* and the negative review *low rating* should not appear in the same sentence connected by the conjunction *and*. These nuances in language usage do impact user's impression of the dialogue system, making the system response rigid and less natural.

To solve this problem, several methods use reinforcement learning (RL) to boost the naturalness of generated responses (Ranzato et al., 2015; Li et al., 2016). However, the Monte-Carlo sampling process in RL is known to have high variance which can make the training process unstable. Li et al. (2015) proposes to use maximum mutual information (MMI) to boost the diversity of language, but this criterion makes exact decoding intractable.

On the other hand, the adversarial training for natural language generation has shown to be promising as the system needs to produce responses indiscernible from human utterances (Rajeswar et al., 2017; Wu et al., 2017; Nie et al., 2018). Apart from the generator, there is a dis-

criminator network which aims to classify system responses from human results. The generator is trained to fool the discriminator, resulting in a min-max game between the two components which boosts the quality of generated utterances (Goodfellow et al., 2014). Due to the discreteness of language, most previous work on adversarial training in NLG apply reinforcement learning, suffering from high-variance problem (Yu et al., 2017; Li et al., 2017; Ke et al., 2019).

In this work, we apply adversarial training to utterance generation in task-oriented dialogues and propose the model AdvNLG. Instead of using RL, we follow Yang et al. (2018) to leverage the Straight-Through Gumbel-Softmax estimator (Jang et al., 2016) for gradient computation. In the forward pass, the generator uses the argmax operation on vocabulary distribution to select an utterance and sends it to the discriminator. But during backpropagation, the Gumbel-Softmax distribution is used to let gradients flow back to the generator. We also find that pretraining the generator for a warm start is very helpful for improving the performance.

To evaluate our model, we conduct experiments on public datasets E2ENLG (Novikova et al., 2017) and RNN-LG (Wen et al., 2016). Our model achieves strong performance and obtains new state-of-the-art results on four datasets. For example, in Restaurant dataset, it improves the best result by 3.6% in BLEU. Human evaluation corroborates the effectiveness of our model, showing that the adversarial training against human responses can make the generated language more accurate and natural.

2 Problem Formulation

The goal of natural language generation module in task-oriented dialogues is to produce system utterances directly issued to the end users (Young, 2000). The generated utterances need to carry necessary information determined by upstream dialogue modules, including the *dialogue act* (DA) and *meaning representation* (MR).

The dialogue act specifies the type of system response (e.g. *inform*, *request* and *confirm*), while the meaning representation contains rich information that the system needs to convey to or request from the user in the form of slot-value pairs. Each slot indicates the information category and each value represents the information content.

Therefore, the training data for the supervised NLG task is $\{x_i = (d_i, r_i), y_i\}_{i=1}^n$, where d_i is the dialogue act, $r_i = \{(s_1, v_1), ..., (s_t, v_t)\}$ is the set of MR slot-value pairs, and y_i is the human-labeled response.

NLG models typically use *delexicalization* during training and inference, replacing slots and values in the utterance with a special token $\langle SLOT\ NAME \rangle$. In this way, the system does not need to generate the proper nouns. Finally, the model substitutes these special tokens with corresponding values when delivering to users.

3 Model

3.1 Generator Model

We use the sequence-to-sequence encoder-decoder architecture (Sutskever et al., 2014) for the response generator G. The input to the encoder is a single sequence x of length m via concatenating dialogue act d and slots and values in the meaning representation r. The target utterance y has n tokens, $y_1, ..., y_n$. Following Zhu et al. (2019), we delexicalize both sequences and surround each sequence with $\langle BOS \rangle$ and $\langle EOS \rangle$ tokens.

Both the encoder and decoder use GRU (Cho et al., 2014) for contextual embedder, and they share the embedding matrix $\mathbf{E}$ to map each token to a fixed-length vector. The final hidden state of the encoder RNN is used as the initial state of the decoder RNN. Moreover, the decoder employs a dot-product attention mechanism (Bahdanau et al., 2014) over the encoder states to get a context vector $\boldsymbol{c}$ at each decoding step.

This context vector $\boldsymbol{c}$ is concatenated with the embedding of the current token and fed into the GRU to predict the next token. The result $\boldsymbol{p}_t = p(y_t|y_1, ..., y_{t-1}; x)$ is the probability distribution of the next token over all tokens in dictionary V.

We use cross entropy as the generator's loss function. Suppose the one-hot ground-truth token vector at the t-th step is $\boldsymbol{y}_t$, then the loss is:

$$\mathcal{L}_{\text{Gen}}(\theta) = -\sum_{t=1}^n \boldsymbol{y}_t^T \log(\boldsymbol{p}_t) \qquad (1)$$

3.2 Adversarial Training

The goal of the adversarial training is to use a discriminator to differentiate between the utterance y' from generator and the ground-truth utterance y.
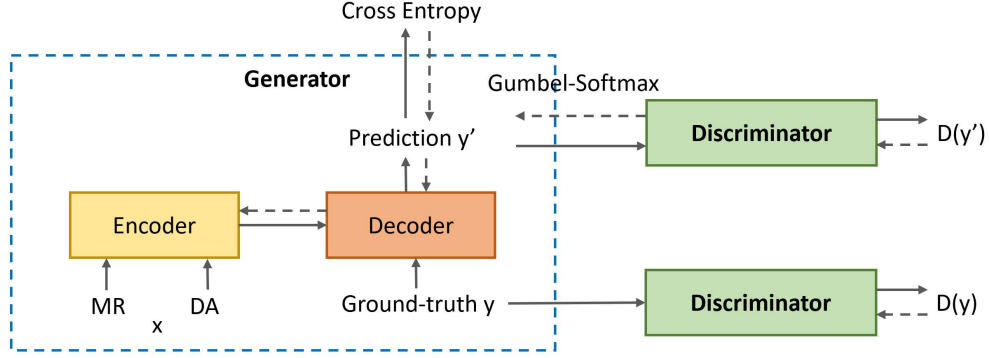
Figure 1: AdvNLG model with generator and discriminator. The dashed arrow is the direction of gradient flow.

We leverage the improved version of generative adversarial network (GAN), Wasserstein-GAN (WGAN) (Arjovsky et al., 2017), in our framework. WGAN designs a min-max game between the generator G and the discriminator D:

$$\min_G \max_D E_{y \sim P_{data}(y)}[D(y)] - E_{y' \sim G(x)}[D(y'))] \quad (2)$$

where $G(x)$ denotes the probability distribution computed by the generator G given input x. The discriminator function D is a scoring function on utterances.

The goal of the generator is to obtain y' as similar as possible to y to fool the discriminator D (the outer-loop min), while D learns to successfully classify generated output y' from the ground-truth y (the inner-loop max), via the scoring function D.

3.2.1 Discriminator Model

For the discriminator, we reuse the embedding matrix $\mathbf{E}$ as the embedder, followed by a bidirectional GRU layer. The last GRU hidden state $\boldsymbol{h}$ is passed through a batch normalization layer and a linear layer to get the final score $D(y)$:

$$\boldsymbol{r} = \text{BatchNorm}(\boldsymbol{h}) \quad (3)$$
$$D(y) = \boldsymbol{W}_3 \boldsymbol{r} + \boldsymbol{b}_3, \quad (4)$$

where $\boldsymbol{W}_3$ and $\boldsymbol{b}_3$ are trainable parameters.

3.2.2 Training

Gradient computation. One problem with adversarial training in language generation is that the token sequence y' sampled from G is discrete, making it impossible to back-propagate gradients from the min-max objective to the generator.

Several previous methods leverage reinforcement learning for gradient computation (Yu et al., 2017; Li et al., 2017). However, the related sampling process can introduce high variance during training. Therefore, we employ the Straight-Through Gumbel-Softmax estimator (Jang et al., 2016; Baziotis et al., 2019). In detail, during the forward pass, at the t-th step, the argmax of the generated word distribution $\boldsymbol{p}_t$ is taken, i.e. greedy sampling. But for gradient computation, the Gumbel-Softmax distribution is used as a differentiable alternative to the argmax operation:

$$\boldsymbol{p}'_{t,i} = \frac{\exp(\log(\boldsymbol{p}_{t,i}) + g_i)/\tau}{\sum_{j=1}^{|V|} \exp(\log(\boldsymbol{p}_{t,j}) + g_j)/\tau}, \quad (5)$$

where $g_1, ..., g_{|V|}$ are i.i.d samples drawn from the Gumbel distribution $G(0,1)$ and τ represents the softmax temperature. Jang et al. (2016) shows that the Gumbel-Softmax distribution converges to the one-hot distribution as $\tau \to 0$ and to the uniform distribution as $\tau \to \infty$. We set $\tau = 0.1$ in all the experiments.

Two-stage Training. We find that the adversarial training does not work well if we optimize both the cross entropy (Eq. 1) and the min-max objective (Eq. 2) from the beginning. However, after we warm up the generator model with only cross entropy loss for several epochs, and then train with the discriminator under both the cross entropy and adversarial objective, the performance is consistently boosted. We argue that during early stages, the generator cannot produce meaningful output, making the discriminator easy to overfit. It's then hard for generator to learn to fool the adversary.

We summarize our model AdvNLG and gradient computation process in Fig. 1.

4 Experiments

We conduct empirical tests on a number of benchmarks for task-oriented dialogues over a variety of domains such as restaurant booking, hotel booking and retail. The datasets include the E2E-NLG

task (Novikova et al., 2017) with 51.4K samples, and the TV, Laptop, Hotel and Restaurant datasets from RNN-LG (Wen et al., 2016), with 14.1K, 26.5K, 8.7K and 8.5K samples respectively. We use BLEU-4 (Papineni et al., 2002) for the automatic metric, computed by the official evaluation scripts from E2E-NLG and RNN-LG.

4.1 Baselines

The baseline systems include TGen (Dušek and Jurčíček, 2016), SC-LSTM (Wen et al., 2015), RALSTM (Tran and Nguyen, 2017), Slug (Juraska et al., 2018), S2S+aug (Nie et al., 2019) and NLG-LM (Zhu et al., 2019). We also implement adversarial training using reinforcement learning in the same way as Li et al. (2017), denoted by RL. The generator in RL is warmed up in the same way as AdvNLG.

4.2 Training Details

In all experiments, the learning rate is 1e-3, the batch size is 20 and the beam width in inference is 10. According to WGAN, the discriminator's parameters are clipped at 0.1. We use RMSprop (Ruder, 2016) as the optimizer. Teacher forcing is used for training the generator, which means that the decoder is exposed to the previous ground-truth token. In warm-up phase, we train the generator for 2 epochs. In E2E-NLG dataset, the generator is updated 5 times before the discriminator is updated once, which is typical in GAN training (Wu et al., 2017). The hyper-parameters above are chosen based on performance on the dev set. Other hyper-parameters like dropout rate, dictionary dimension and RNN hidden size are the same with Table 3 in Zhu et al. (2019).

For baseline models, we implemented NLG-LM (Zhu et al., 2019) and reproduced its results. We obtain the prediction results of Slug (Juraska et al., 2018) from its open-source website.

4.3 Results

As shown in Table 2, our model AdvNLG achieves new state-of-the-art results on TV, Laptop, Hotel and Restaurant datasets, improving previous best results by 0.8%, 3.8%, 0.6% and 3.6%. Statistical tests show that this advantage is statistically significant with p-values smaller than 0.05. Our model also obtains results on par with NLG-LM on E2ENLG. We show some prediction examples in Table 4. Generally, with adversarial training, the generated output can group information from

Model	E	TV	L	H	R
TGen	0.659	/	/	/	/
Slug	0.662	0.529	0.524	/	/
SCLSTM	/	0.527	0.512	0.848	0.752
RALSTM	/	0.541	0.525	0.898	0.779
S2S+aug	0.665	/	/	/	/
NLG-LM	**0.684**	0.617	0.586	0.939	0.795
AdvNLG	**0.683**	0.625*	0.624*	0.945*	0.831*
RL	0.674	0.605	0.606	0.932	0.796
-Adv.	0.671	0.618	0.564	0.931	0.753
-2 stages	0.662	0.621	0.557	0.932	0.782

Table 2: BLEU scores on E2ENLG (**E**), TV, Laptop (**L**), Hotel (**H**) and Restaurant (**R**) testset. *: means the result is statistically significant with p-value<0.05. **-Adv.** means we only train the generator, with cross entropy loss. **-2 stages** means that both the generator and discriminator are trained together from scratch.

the same category together, while placing positive and negative aspects (e.g. family-friendly and expensive) in different sentences.

Ablation Study. The bottom section of Table 2 shows that adversarial training can boost performance by 0.7% to 7.8%. Our proposed two-stage training is also very beneficial. If both generator and discriminator are trained from scratch, the result drops significantly. RL-based adversarial training achieves mixed results. On TV dataset, it even hurts the performance. We attribute this to the high variance and instability in training.

Model	Naturalness	Accuracy
Slug	2.51 (0.48)	2.89 (0.36)
NLG-LM	2.52 (0.46)	2.84 (0.41)
AdvNLG	**2.84*** (0.27)	**2.97*** (0.17)
-Adv.	2.45 (0.53)	2.63 (0.58)

Table 3: Average human evaluation ratings (1-3, 3 is best) for naturalness and accuracy of output generated by different models. Standard deviation is shown in parenthesis. *: the p-value is smaller than 0.01.

4.4 Human Evaluation

We randomly sample 100 data-text pairs from the test set of E2ENLG. We then ask 3 labelers to judge the accuracy and naturalness of the utterances generated by Slug, NLG-LM, AdvNLG with and without adversarial training. The accuracy measures how precise the utterance expresses

the dialogue act and meaning representation. The naturalness is measured by how likely the labeller thinks the utterance is spoken by a real human. In addition to the model output, each labeler is also given the meaning representation and the ground truth. The labelers need to give an integer rating from 1 to 3 (3 being the best) for each criterion.

Table 3 shows that our AdvNLG model has an apparent lead in both naturalness and accuracy, and the paired t-test shows that the result is statistically significant with p-value smaller than 0.01. And our ablation model -Adv. achieves the lowest score, proving that adversarial training can boost both naturalness and accuracy.

5 Conclusion

In this paper, we propose adversarial training using the Straight-Through Gumbel-Softmax estimator in NLG for task-oriented dialogues. We also propose a two-stage training scheme to further boost the gain in performance. Experimental results show that our model, AdvNLG, consistently outperforms state-of-the-art models in both automatic and human evaluations.

In the future, we plan to apply this method to other conditional generation tasks, e.g. produce a natural utterance containing a given list of keywords.

Acknowledgement

References

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

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Christos Baziotis, Ion Androutsopoulos, Ioannis Konstas, and Alexandros Potamianos. 2019. Seqˆ3: Differentiable sequence-to-sequence-to-sequence autoencoder for unsupervised abstractive sentence compression. *arXiv preprint arXiv:1904.03651*.

Adam Cheyer and Didier Guzzoni. 2014. Method and apparatus for building an intelligent automated assistant. US Patent 8,677,377.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Ondřej Dušek and Filip Jurčíček. 2016. Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings. *arXiv preprint arXiv:1606.05491*.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. pages 2672–2680.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Juraj Juraska, Panagiotis Karagiannis, Kevin K Bowden, and Marilyn A Walker. 2018. A deep ensemble model with slot alignment for sequence-to-sequence natural language generation. *arXiv preprint arXiv:1805.06553*.

Pei Ke, Fei Huang, Minlie Huang, and Xiaoyan Zhu. 2019. Araml: A stable adversarial training framework for text generation. *arXiv preprint arXiv:1908.07195*.

Irene Langkilde and Kevin Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 704–710. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

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*.

Feng Nie, Jin-ge Yao, Jinpeng Wang, Rong Pan, and Chin-Yew Lin. 2019. A simple recipe towards reducing hallucination in neural surface realisation. pages 2673–2679.

Weili Nie, Nina Narodytska, and Ankit Patel. 2018. Relgan: Relational generative adversarial networks for text generation.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017. The e2e dataset: New challenges for end-to-end generation. *arXiv preprint arXiv:1706.09254*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Sai Rajeswar, Sandeep Subramanian, Francis Dutil, Christopher Pal, and Aaron Courville. 2017. Adversarial generation of natural language. *arXiv preprint arXiv:1705.10929*.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*.

Sebastian Ruder. 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Van-Khanh Tran and Le-Minh Nguyen. 2017. Natural language generation for spoken dialogue system using rnn encoder-decoder networks. *arXiv preprint arXiv:1706.00139*.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Lina M Rojas-Barahona, Pei-Hao Su, David Vandyke, and Steve Young. 2016. Multi-domain neural network language generation for spoken dialogue systems. *arXiv preprint arXiv:1603.01232*.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.

Lijun Wu, Yingce Xia, Li Zhao, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2017. Adversarial neural machine translation. *arXiv preprint arXiv:1704.06933*.

Zichao Yang, Zhiting Hu, Chris Dyer, Eric P Xing, and Taylor Berg-Kirkpatrick. 2018. Unsupervised text style transfer using language models as discriminators. pages 7287–7298.

Steve J Young. 2000. Probabilistic methods in spoken–dialogue systems. *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 358(1769):1389–1402.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2019. Multi-task learning for natural language generation in task-oriented dialogue.

MR	name[Wildwood], eatType[restaurant], food[Indian], area[riverside], familyFriendly[no], near[Raja Indian Cuisine]
Ref.	Located in the riverside area near the Raja Indian Cuisine, Wildwood offers Indian food and a restaurant. It is not family friendly.
AdvNLG	Wildwood is an Indian restaurant in the riverside area near Raja Indian Cuisine. It is not family friendly.
-Adv.	Wildwood is a restaurant providing Indian food. It is located in the riverside. It is near Raja Indian Cuisine.
NLG-LM	Wildwood is a restaurant providing Indian food. It is located in the riverside. It is near Raja Indian Cuisine.
Comment	Only AdvNLG places the two pieces of location information "riverside" and "near Raja Indian Cuisine" together, which is aligned with human language patterns.
MR	name[The Cricketers], eatType[restaurant], food[English], priceRange[high], customer rating[1 out of 5], area[city centre], familyFriendly[yes], near[Café Rouge]
Ref.	The Cricketers, an English restaurant located near Café Rouge in the city centre, offers food at high price range. Although it has a customer rating of 1 out of 5, it also is children friendly.
AdvNLG	The Cricketers is a child friendly English restaurant in the city centre near Café Rouge. It has a high price range and a customer rating of 1 out of 5.
-Adv.	The Cricketers is a restaurant located in the city centre near Café Rouge. It is a high priced restaurant that serves English food. It is rated 1 out of 5 and is children friendly.
NLG-LM	The Cricketers is a high priced English restaurant located in the city centre near Café Rouge. It has a customer rating of 1 out of 5 and is child friendly.
Comment	AdvNLG model naturally put the negative aspects like "high price" and "rating 1 out of 5" together with conjunction "and", whereas both -Adv. and NLG-LM juxtapose negative aspect (low customer rating) and positive aspect (kid-friendly) in one sentence, which appears contradictory.
MR	name[The Plough], eatType[restaurant], food[Chinese], priceRange[cheap], area[riverside], familyFriendly[yes], near[Raja Indian Cuisine]
Ref.	The Plough is a cheap Chinese restaurant located riverside by Raja Indian Cuisine. It is a family friendly establishment.
AdvNLG	The Plough is a cheap Chinese restaurant in the riverside area near Raja Indian Cuisine. It is family friendly.
-Adv.	The Plough is a cheap family friendly restaurant that serves Chinese food. It is located in the riverside area near Raja Indian Cuisine.
NLG-LM	The Plough is a restaurant providing Chinese food in the cheap price range. It is located in the riverside. It is near Raja Indian Cuisine.
Comment	AdvNLG places "Chinese" immediately before "restaurant", and this is in line with the human reference. And NLG-LM model has two less connected sentences at the end.

Table 4: Example of predictions on E2E-NLG by reference, NLG-LM model and our model AdvNLG with and without adversarial training. As E2E-NLG only has *inform* dialogue act, we show the meaning representation (MR).