ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414915

TOWARDS EFFICIENTLY DIVERSIFYING DIALOGUE GENERATION VIA EMBEDDING AUGMENTATION

Yu Cao†, *Liang Ding*†*, *Zhiliang Tian*§, *and Meng Fang*‡

† School of Computer Science, The University of Sydney, Australia, ‡ Tencent Robotics X, China
§ Department of CSE, The Hong Kong University of Science and Technology, Hong Kong

ABSTRACT

Dialogue generation models face the challenge of producing generic and repetitive responses. Unlike previous augmentation methods that mostly focus on token manipulation and ignore the essential variety within a single sample using hard labels, we propose to promote the generation diversity of the neural dialogue models via soft embedding augmentation along with soft labels in this paper. Particularly, we select some key input tokens and fuse their embeddings together with embeddings from their semantic-neighbor tokens. The new embeddings serve as the input of the model to replace the original one. Besides, soft labels are used in loss calculation, resulting in multi-target supervision for a given input. Our experimental results on two datasets illustrate that our proposed method is capable of generating more diverse responses than raw models while remains a similar n-gram accuracy that ensures the quality of generated responses.

Index Terms— Dialogue generation, diversity, data augmentation, embedding, natural language processing

1. INTRODUCTION

Dialogue generation is important in AI applications, in which a model can generate responses for user-issued dialogue history. Former dialogue datasets make the end-to-end training of deep neural models possible, such as DailyDialog [1], PersonaChat [2], etc., and deep neural networks including Seq2seq with attention [3], Transformers [4] have already shown their capability in generating conversation replies. Dialogue generation can be regarded as a **many-to-many** (combination of one-to-many and many-to-one) problem as one specific response may be reasonable for multiple histories and vice versa. However, current models tend to produce generic sentences such as "I don't know", caused by the current inherent deterministic training objective as well as insufficient diversity and limited quality of current datasets [5, 6].

Generating more various dialogue responses remains a hard task. In order to tackle it, some researchers try to refine the training objective with extra constraint items [5] or modify the criterion that encourages models to decode more diverse words [7, 8, 9, 10]. However, such models are more likely to generate ungrammatical or uncorrelated responses and the superiority of constructed objectives to cross-entropy loss remains unknown. Another kind of approach enhances the training data directly. Data filtering is commonly used in machine learning and extended to dialogue generation by removing samples with generic responses based on entropy [11] or the predictions of a Seq2seq model [12]. Obtaining more training samples is also a possible solution. Word replacement extends the data scales by randomly replacing original tokens with others based on vocabulary distributions [13, 14]. In addition, new augmented sentences can also be obtained via a learnable model that even keeps interaction between the generation model and both of them will be trained jointly [15]. But these methods merely include more training data and still bridge a one-to-one mapping between input and response within a single sample. And they also add great extra computational load. So it is necessary to propose a simple and effective method regarding the essence of this problem.

In this paper, we propose an embedding augmentation method for dialogue models, where not only the training objective encourages various output, but also soft embeddings enhance the diversity within a single sample. It is inspired by the recent success of the mixup approach that combines training pairs of samples and labels convexly into a single one [16]. Particularly, the original embeddings of tokens in the training samples will be randomly replaced by augmented ones, which are mixtures of the raw one and several semantically similar embeddings conditioned on their distribution. And these selected positions will use soft labels instead of hard one-hot vectors. Such a mechanism ensures that the model can learn a *soft word distribution* rather than a fixed word in both input and output, which is consistent with the multi-source and multi-target purpose, benefiting the generating diversity without using any additional sample. Compared to previous similar work [17, 18], we use both soft embedding and soft labels for a more flexible training process. A compact in-domain similar token prediction model is utilized instead of a deep language model, realizing a more efficient

*This work was done when Yu Cao and Liang Ding were interning at Tencent AI Lab. Code is available: `https://github.com/caoyu-noob/embedding_augmentation`.

augmentation with a less computational cost.

To verify our method, we conduct experiments on two dialogue datasets, PersonaChat and DailyDialog, using two base models, Seq2seq and transformer. The experimental results show that our method can obtain remarkable diversity improvement as well as n-gram accuracy boost in terms of auto metrics compared to models without embedding augmentation or using token-level replacement. We also compare it with a BERT-based embedding augmentation method, demonstrating they have close performance while the former one is much faster and proving the efficiency of our method.

2. METHOD

2.1. Background and motivations

Generally speaking, given a dialogue history $H = (h_1, h_2, ..., h_N)$ and a corresponding response $R = (r_1, r_2, ..., r_M)$, current neural dialogue generation models learn the conditional probability $p(r_1, r_2, ..., r_M | h_1, h_2, ..., h_N)$ which is similar to machine translation. But as discussed before, dialogue tasks should be many-to-many that differs from machine translation who has higher certainty. Nevertheless, there is usually one specific R that corresponds to a given H in most datasets. Although one-to-many or many-to-one cases exist, most of them are generic sentences having no contribution to the variety of generated replies [11]. Thus existing samples may provide the wrong guidance for the task goal.

Some existing augmentation methods generate new samples by replacing tokens in original sentences with others [14, 17]. But such a hard replacement will also magnify the error if inappropriate candidate tokens are introduced. Although soft word distribution is employed in [18], it still uses hard labels for loss calculation. Such augmentation only ensures a many-to-one learning objective for the model, differing from our desired target. These motivate us to use soft word embeddings and soft labels in augmentation. The use of fused embedding can lower the influence of unbefitting tokens as partial original information still remains in the embedding. While the latter one can bring randomness and multiple targets during training to further diversify the generation.

2.2. Embedding augmentation

Inspired by the above intuition, we provide an embedding augmentation method, whose framework is shown in Fig. 1. An encoder-decode model is used for generation. The history H is encoded to get its hidden state as the initial state of decoder. The model is trained with supervision from the response R. Embeddings of selected tokens in H and R will be replaced with augmented ones from distributions of their semantic neighbors. These distributions are also used as labels.

We first select target tokens for augmentation with some exceptions, which is indicated by the red font in Fig. 1. Since special tokens including **commas**, **articles** (a/an/the) and

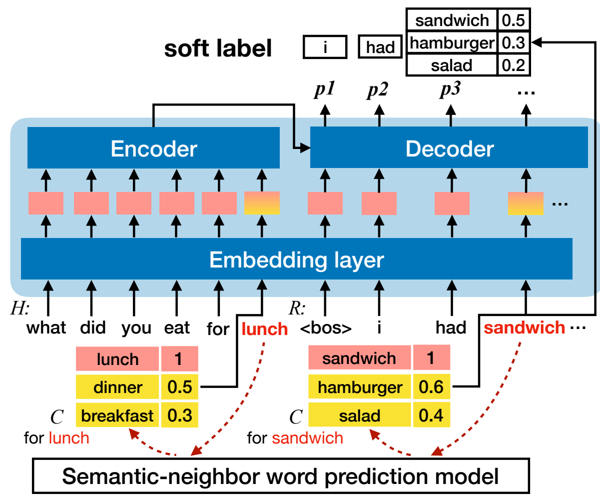

Fig. 1. Framework of our embedding augmentation method.

prepositions (in/on, etc.) are usually unique, replacing their embedding has slight benefit for diversity or even causes semantically degeneration. They are not considered as target tokens. The rest tokens are randomly selected with augmentation ratio ρ, resulting in a set target set $\mathcal{T} = (t_1, ..., t_{n+m})$ with n tokens from H and m tokens from R.

Then a semantic-neighbor word prediction model is utilized to predict the candidate words with the most similar meaning for each token in $\mathcal{T}$. It prevents semantic ambiguity when mixing the embeddings from these candidates together. A most intuitive method is using a trained large-scale language model such as BERT [19] to contextually predict candidates. But considering the computational efficiency of our method, we use *fastText* model here [20], and the effectiveness of using this model will be later illustrated in the experiment. It is based on a continuous-bag-of-word frame which is fast to train. Given a word, it will output a series of candidates $c_i \in \mathcal{V}$ with confidence scores $s_i \in (0, 1)$. We will train the fastText model using the same corpus as the training data for the dialogue model, so as to get a consistent vocabulary $\mathcal{V}$ as well as in-domain word distribution. To ensure the quality of augmentation, only candidates with top-*k* confidence remain, and their scores must satisfy $s_i \geq \tau$ where τ is a threshold.

One target token t and its semantic neighbors can form a soft word set $\mathcal{C} = \{(c_0, s_0), (c_1, s_1), ..., (c_k, s_k)\}, c_0 = t$, here $s_0 = 1$ keeps a higher proportion of the raw information as we believe it is more reliable than predicted candidates. c_j in this set can then be transformed into a set of new representations $\{e_0, e_1, ..., e_k\}, e_j \in \mathbb{R}^d$ via the embedding layer. The fused embedding e^f of them for t can be obtained via

$$e^f = \sum\nolimits_{j=0}^{j=k} p(c_j) \cdot e_j, \text{ where } p(c_j) = s_j / \sum\nolimits_{l=0}^{l=k} s_l. \quad (1)$$

It can be expected as an augmented soft embedding based on the distribution of the original token and its semantic neighbors to replace the raw embedding. The distribution $p(c_j)$ is calculated using scores from the neighbor prediction model.

It is worth pointing out that no external knowledge is included as we just depend on the original dataset, which is different from some previous works [**?**, 17]. And our method can be easily extended to other different base models as the only model-level modification is the embedding layer.

2.3. Model training

The cross-entropy loss will be employed as the optimization objective. Given the generation logits $g \in \mathbb{R}^{|\mathcal{V}|}$ after softmax function for current position i in label (response), if r_i is selected as the target augmentation token, the loss for r_i is

$$L_i = -\sum_{j=0}^{j=k} p(c_{ij}) \cdot log\ g(c_{ij}), \tag{2}$$

where c_{ij} is a token from the candidate set $\mathcal{C}$ for r_i, and $p(c_j)$ is the distribution among candidates as (1), $g(c_j)$ denotes getting the corresponding value in vector g for c_j. If r_i is not an augmentation token, the training loss becomes $L_i = -log\ g(r_i)$, which is the same as the previous models.

Moreover, the augmentation operation is executed alternately along with the normal training procedure without augmentation in each step. This aims to guarantee a correct learning direction by original samples and prevent possible continuous error propagation from augmented embeddings.

3. EXPERIMENT

3.1. Settings

Two public dialogue datasets, PersonaChat [2] and DailyDialog [1] are involved in our experiment, containing 131k/ 7.8k/ 7.5k and 42.6k/ 3.9k/ 3.7k for training/validation/test set respectively. The former one contains persona profiles as auxiliary information while the latter one contains topic information. We will concatenate it with the original dialog history as the input H, using a special token to distinguish each part.

We consider two base end-to-end models, Seq2Seq with Attention (**Seq2seq**) [3] using 1-layer bidirectional GRU as the encoder and decoder, and transformer model (**transfo**) [4] in which the encoder and decoder both have 6 layers and 4 heads in attention. 300d GloVe [21] is used to initiate the embedding layer. The hidden dimension for both models is set as 300. Each model on each dataset is trained via Adam optimizer with batch size 256 for 50 epochs(PerosnaChat)/ 30 epochs(DailyDialog). During decoding, beam search with size 3 is applied to the model with the best PPL. Besides, a fastText model will be trained on each dataset for 100 epochs using the 'cbow' method (a config of fastText means continuous bag-of-word), then used as the neighbor prediction model correspondingly. In augmentation, the augmentation ratio ρ is 0.4 for PersonaChat and 0.5 for DailyDialog, the score threshold τ is 0.4, and top-ranking number k is 5.

Except for our embedding augmentation (**-EA**), we also consider two baselines: 1) the original model without augmentation; 2) a replacement baseline (**-rep**) in which a token is directly replaced by the most similar one in its neighbor candidates. Besides, one variant is included that uses the original BERT to replace fastText (**-BERT**) for augmentation candidates prediction while the rest parts remain the same.

We use multiple automatic metrics to evaluate the performance. **BLEU** [22], **METEOR** [23] and **NIST-4** [24] are used for n-gram accuracy of generated replies. Following metrics are considered for measuring diversity, 1) **Ent-n** [25] calculate the entropy based on the appearance probabilities of each n-gram in all predictions; 2) **Dist-n** (distinct) [5] is defined as the ratio of unique n-grams over all n-grams in all generated sentences; 3) **Sen-n** is the average of sentence-level **Dist-n**(n=1,2,3). Besides, generation perplexity (PPL) and average length (avg.len) of responses are also provided.

3.2. Main results

The augmentation results along with two baselines, the original model and replacement method, on two datasets are shown in Table 1. It can be found that our method outperforms them under most conditions in both n-gram accuracy and diversity evaluation. Compared to the original models, our method (EA) can significantly promote the n-gram diversity of responses and slightly benefit n-gram accuracy under most conditions while remaining equivalent under other cases. In other words, EA encourages models to generate diverse texts without the cost of their quality. In contrast, the replacement method usually degrades the generation accuracy despite it is also beneficial for more various replies.

When comparing between base models, EA seems to have more effect on transformer than Seq2seq. The reason is that more data is needed to train a robust transformer than RNN due to its depth, and our method pose a similar effect. It can also be observed that the PPL values of both EA and replacement can result in a noticeable PPL decrease for transformer models, but it is not such a case for Seq2seq models. It means transformers may not be well trained using current raw data.

We also evaluate the performance of Seq2seq / transformer-BERT to better prove the merits of using fastText as a semantic-neighbor word prediction model. The performance of them is shown in Table 2 in terms of accuracy metrics, Ent, Dist, and Sen(the average of Ent/Dist/Sen-n and n=1, 2, 3). EA and BERT variant show equivalent performance on both base models, because BERT embeddings without tuning on a specific domain can even perform worse than a universal embedding [26]. However, the huge BERT model will increase the computational load since the training time of our EA is approximately 1.98× time of raw models, while it is 5.87× for BERT variant and 1.37× for replacement according to experiments. Obviously, it is better to select fastText rather than BERT for augmentation candidates prediction.

	Model	PPL	avg.len	BLEU	MET	N-4	Ent-1	Ent-2	Ent-3	Dist-1	Dist-2	Dist-3	Sen-1	Sen-2	Sen-3
PersonaChat	Seq2seq	36.016	8.462	2.508	7.636	1.003	4.114	5.992	7.326	1.811	8.646	20.286	**89.547**	96.609	98.211
	Seq2seq-rep	36.814	8.579	2.685	8.135	1.168	**4.156**	6.115	7.509	**1.888**	9.227	21.032	87.628	96.079	98.756
	Seq2seq-EA	**35.368**	8.651	**2.729**	**8.206**	**1.274**	4.137	**6.159**	**7.564**	1.802	**9.653**	**21.891**	88.547	**96.684**	**98.847**
	transfo	39.882	8.602	2.839	7.840	1.107	4.112	5.570	6.313	1.532	5.358	10.059	**92.088**	97.432	98.390
	transfo-rep	35.993	8.620	3.031	8.121	1.175	4.240	5.844	6.696	**1.771**	6.787	12.733	90.793	96.963	98.227
	transfo-EA	**34.480**	8.652	**3.450**	**8.414**	**1.204**	**4.269**	**5.963**	**6.821**	1.753	**6.809**	**12.927**	91.387	**97.833**	**98.756**
DailyDialog	Seq2seq	53.707	7.008	1.108	**5.471**	**0.291**	**4.515**	6.579	**7.691**	4.956	18.038	**35.422**	92.098	96.953	**92.725**
	Seq2seq-rep	54.944	6.560	1.008	5.140	0.162	4.436	6.411	7.425	4.980	18.524	33.857	**94.923**	**98.116**	88.474
	Seq2seq-EA	**53.541**	6.804	**1.214**	5.452	0.217	4.455	**6.648**	7.657	**5.095**	**19.459**	34.543	94.359	97.941	92.691
	transfo	77.952	7.009	1.005	5.032	0.202	3.923	5.375	6.087	2.296	7.692	12.925	94.076	97.877	87.817
	transfo-rep	74.989	6.732	1.051	5.025	**0.207**	**4.089**	5.576	**6.374**	**3.033**	**9.434**	16.090	94.820	98.171	89.431
	transfo-EA	**74.020**	6.835	**1.097**	**5.106**	0.198	4.052	**5.631**	6.365	2.989	9.263	**16.658**	**95.280**	**98.187**	**90.608**

Table 1. Automatic evaluation results on two datasets (% for BLEU, Dist-n and Sen-n). MET: METEOR, N-4: NIST-4.

Model	ΔPPL	ΔBLEU	ΔN-4	ΔEnt	ΔDist	ΔSen
Seq2seq-BERT	*1.212*	0.050	*-0.012*	0.032	*-0.130*	*-0.738*
transfo-BERT	1.688	*-0.108*	*-0.023*	*-0.116*	*-0.091*	*-0.743*
Seq2seq-BERT	-1.885	*-0.155*	*-0.082*	*-0.046*	0.597	*-0.408*
transfo-BERT	*2.860*	*-0.011*	0.071	0.104	1.057	*-0.739*

Table 2. Performance value difference (Δ) when the BERT variant of our method compared to our original one. Upper: PersonaChat, lower:DailyDialog. (Italic: -BERT is worse)

Model	PPL	BLEU	N-4	Ent	Dist	Sen
Seq2seq-EA	53.541	1.214	0.217	6.253	19.699	95.057
w/o soft label	55.454	0.861	0.098	6.065	19.160	94.597
w/o history aug.	59.684	0.956	0.134	6.024	17.606	94.105
transfo-EA	74.020	1.097	0.198	5.349	9.637	94.692
w/o soft label	72.135	0.879	0.100	5.241	9.182	94.127
w/o history aug.	77.499	0.985	0.139	5.221	9.536	94.381

Table 3. Results of the ablation study on DailyDialog dataset.

3.3. Ablation study

We also conduct the ablation study including the following variants: 1) w/o soft label: we only use the original hard labels during training which is the same as the previous models; 2) w/o history augmentation(aug.): we only applying embedding augmentation to response R. The related results of two models on DailyDialog dataset is shown in Table 3. We easily find out the contribution of both parts according to the performance degeneration under most conditions.

3.4. The influence of augmentation ratio

To better illustrate the influence of different augmentation ratio ρ, we also tried different ρ values from 0 to 0.8. The BLEU score (for n-gram accuracy), Dist, and Sen values (for diversity) of transformer model on two datasets vary with ρ is shown in Fig. 2. In the beginning, both BLEU score and diversity metrics tend to increase slowly. But after a middle value (0.4 on PersonaChat and 0.5 on DailyDialog), BLEU scores begin to drop while generation diversity is also becoming saturated. A higher ratio than these peak values will not benefit the performance but merely increase the training time as more neighbor prediction and embedding fusion operations are needed. That is the reason why we choose these ρ values.

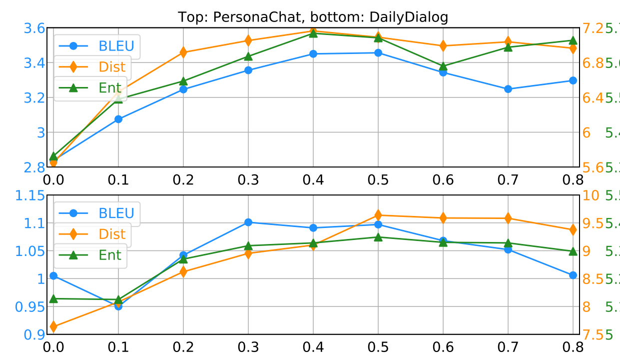

Fig. 2. The performance of n-gram accuracy and diversity of transformer-EA varies with different augmentation ratio ρ.

4. CONCLUSION

Considering the many-to-many essence of the dialogue generation problem, we propose an efficient embedding augmentation method aiming to promote the response diversity. Different from previous work, our manipulation is based on embedding level instead of token level by transforming a soft word, which is a distribution between the original word and its semantic neighbors, into a fused embedding to randomly replace the original one. We also use such a distribution as a soft label to realize our multi-target training goal. The experimental results demonstrate that our method can effectively diversify the produced replies from both raw Seq2seq and transformer models with even slight improvement in accuracy, and it is also superior to replacement-based methods.

5. REFERENCES

[1] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu, "Dailydialog: A manually labelled multi-turn dialogue dataset," in *EMNLP*, 2017, pp. 986–995.

[2] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston, "Personalizing dialogue agents: I have a dog, do you have pets too?," in *ACL*, 2018, pp. 2204–2213.

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

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

[5] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan, "A diversity-promoting objective function for neural conversation models," in *HLT-NAACL*, 2016.

[6] Yuanlong Shao, Stephan Gouws, Denny Britz, Anna Goldie, Brian Strope, and Ray Kurzweil, "Generating high-quality and informative conversation responses with sequence-to-sequence models," in *EMNLP*, 2017, pp. 2010–2219.

[7] Gakuto Kurata, Bing Xiang, and Bowen Zhou, "Labeled data generation with encoder-decoder lstm for semantic slot filling.," in *INTERSPEECH*, 2016, pp. 725–729.

[8] Ilia Kulikov, Alexander Miller, Kyunghyun Cho, and Jason Weston, "Importance of search and evaluation strategies in neural dialogue modeling," in *Proceedings of the 12th International Conference on Natural Language Generation*, 2019, pp. 76–87.

[9] Jiwei Li, Will Monroe, and Dan Jurafsky, "A simple, fast diverse decoding algorithm for neural generation," *arXiv preprint arXiv:1611.08562*, 2016.

[10] Yiping Song, Zhiliang Tian, Dongyan Zhao, Ming Zhang, and Rui Yan, "Diversifying neural conversation model with maximal marginal relevance," in *IJCNLP*, 2017, pp. 169–174.

[11] Richard Csaky, Patrik Purgai, and Gabor Recski, "Improving neural conversational models with entropy-based data filtering," *arXiv preprint arXiv:1905.05471*, 2019.

[12] Jiwei Li, Will Monroe, and Dan Jurafsky, "Data distillation for controlling specificity in dialogue generation," *arXiv preprint arXiv:1702.06703*, 2017.

[13] Ziang Xie, Sida I Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y Ng, "Data noising as smoothing in neural network language models," *arXiv preprint arXiv:1703.02573*, 2017.

[14] Marzieh Fadaee, Arianna Bisazza, and Christof Monz, "Data augmentation for low-resource neural machine translation," pp. 567–573, 2017.

[15] Juntao Li, Lisong Qiu, Bo Tang, Dongmin Chen, Dongyan Zhao, and Rui Yan, "Insufficient data can also rock! learning to converse using smaller data with augmentation," in *AAAI*, 2019, vol. 33, pp. 6698–6705.

[16] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[17] Sosuke Kobayashi, "Contextual augmentation: Data augmentation by words with paradigmatic relations," *arXiv preprint arXiv:1805.06201*, 2018.

[18] Fei Gao, Jinhua Zhu, Lijun Wu, Yingce Xia, Tao Qin, Xueqi Cheng, Wengang Zhou, and Tie-Yan Liu, "Soft contextual data augmentation for neural machine translation," in *ACL*, 2019, pp. 5539–5544.

[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2018, pp. 4171–4186.

[20] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[21] Jeffrey Pennington, Richard Socher, and Christopher D Manning, "Glove: Global vectors for word representation," in *EMNLP*, 2014, pp. 1532–1543.

[22] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "Bleu: a method for automatic evaluation of machine translation," in *ACL*, 2002, pp. 311–318.

[23] Satanjeev Banerjee and Alon Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005, pp. 65–72.

[24] Zhiliang Tian, Wei Bi, Dongkyu Lee, Lanqing Xue, Yiping Song, Xiaojiang Liu, and Nevin L Zhang, "Response-anticipated memory for on-demand knowledge integration in response generation," in *ACL*, pp. 650–659.

[25] Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan, "Generating informative and diverse conversational responses via adversarial information maximization," in *NIPS*, 2018.

[26] Nils Reimers and Iryna Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *EMNLP-IJCNLP*, 2019, pp. 3973–3983.