Modelling Personalized Dialogue Generation in Multi-Party Settings

Rohan Kumar
Electrical Engineering
Indian Institute of Technology, Patna
Patna, India
rohan.ee17@iitp.ac.in

Dushyant Singh Chauhan
Computer Science and Engineering
Indian Institute of Technology, Patna
Patna, India
1821CS17@iitp.ac.in

Gaël Dias
Department of Computer Science
Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC
14000, Caen, France
gael.dias@unicaen.fr

Asif Ekbal
Computer Science and Engineering
Indian Institute of Technology, Patna
Patna, India
asif@iitp.ac.in

***Abstract*—Naive dialogue generation systems do not have the ability to generate distinguishable utterances while replying to different speakers, as they do not take into consideration whom the speaker is. To overcome this situation, we present an end-to-end deep learning model that relies on a person-specific embedding (persona) to generate adequate responses in multi-party conversations. In particular, the persona contains information about how a person behaves in the conversational multi-party setting. Empirical results on the Multi-Domain Wizard-of-Oz (MultiWoz) data set show the efficacy of our approach over the existing state-of-the-art systems, and show that our approach efficiently generates person-specific utterances.**

I. Introduction

Modelling conversational agents with human-like abilities has always been a challenging task. The development of such systems means a step closer to general artificial intelligence and has recently attracted the attention of a wide range of studies [20]. Conversational systems find applications in domains, such as health [25], smart home control [4], or service systems [23], but also in the open domain [21].

The generation process poses serious challenges as context-awareness is a key factor. In a two-speaker dialogue system, the utterances generated by the agent depend on the whole conversation history, which in itself is challenging. But in a multi-party setting, the agent should also know the person to whom he will generate the dialogue. Indeed, depending on the addressee, the sentence production might differ both in its form and/or semantics [16].

To overcome this situation, recent studies have introduced the notion of person-specific embeddings also called persona [9], [11], where a dyadic speaker addressee model captures properties of interactions between two interlocutors. In particular, the persona avoids the creation of specific multiple agent/addressee models, i.e. one for each pair of speakers. The biggest challenge faced while generating dialogues for a multi-party conversation vs. generating dialogues for a two person or dyadic conversation is the liberty of choosing one speaker as the *input* and the other speaker as the *output*. If we try to extend this idea to a multi-party setting we would require a separate *text generative agent* for each speaker. However, this would prove to be very resource consuming and hence not scalable. A better alternative is to have the information of both speakers and the their social interaction traits and use them to generate person-specific response using the only the single model.

To achieve this said information, we use a similar technique used for generating word embeddings, i.e obtaining the features or traits of the word in vectorized form. We extend this technique to generate persona embedding by generating a vectorization of social interaction traits or persona of a speaker.

The main contributions of our research are as follows: **a).** we present a deep learning approach to create the person-specific embedding (persona); **b).** we discuss different use cases of our persona for a wide range of baseline models, including sequence-to-sequence, transformers and generative adversarial networks architectures; **c).** we illustrate state-of-the-art results for dialogue generation in a multi-party scenario over the Multi-Domain Wizard-of-Oz data set [1].

II. Related Work

Due to a more realistic nature, multi-party dialogue generation has recently attracted attention [6], [7], [9], [16], [28]. Nevertheless, most progresses in dialogue systems have been proposed in two-party settings. In particular, transformer-based dialogue generation architectures such as GPT-2 [19] have been very successful as a robust end-to-end natural language generation system. In mutli-party settings, [15] recently proposed one such transformer-based dialogue system. In other recent developments in dialogue generation trends, we have also seen improvements in performance by employing adversarial training [10]. Within the multi-party setting, authors in [14], [24] proposed multi-turn dialogue generation models

Dialogue
Hello, I have been robbed. Can you please help me get in touch with the police?
Parkside Police Station is in Parkside, Cambridge. Their number is 01223358966. Anything else I can do for you?
Can I please have the postcode as well?
The postcode for the Parkside Police Station is CB11JG. Can I help you with anything else?
Was Parkside the address of the police station? If not, can I have the address please?
Yes, Parkside is the address.
Thank you that will be all for now.
Great. Thank you for contacting Cambridge Towninfo Centre.
You were great. Goodbye.
We are happy to help. Have a good day!

TABLE I
EXAMPLE OF CONVERSATION *MultiWoz*.

using generative adversarial networks (GANs) combined with encoder-decoder architectures.

For the "MultiWoz: wizard of oz" [1] dataset, one of the previous works [29] proposes a reinforcement learning framework that treats the action spaces of the dialog agent as the latent variables for end-to-end dialogue generation. Meanwhile [13] proposes a scalable and accurate neural dialogue state tracking mode, which uses global modules to share parameters between estimators for different types of dialogue states. [8] proposed a deep learning based scalable framework for dialog generation by learning to estimate probability distribution at every dialog turn.

In order to capture the personality of the speaker in a multi-party conversation, [9] proposed to build an embedding that catches individual characteristics such as background information and speaking style, as well as interactions properties between addressees. In order to generate a dialogue utterance, a linear combination of persona embeddings is concatenated to the latent representation. While this approach yields good results with the recurrent neural network architecture they adopt, it poses a new challenge of incorporating such one dimension persona embedding with a multi-dimension latent representation as obtained in the case of transformers.

In comparison to these existing research works, our proposed approach aims at creating a scalable solution to multi-party dialogue generation by creating a social-persona embedding for every speaker in a conversation. Such a built persona is then used to create addressee-differentiated utterances in an attempt to capture interpersonal and behavioral relationships between the speaker and the addressee.

III. PROPOSED METHODOLOGY

Similar to word embeddings we propose a set of persona embeddings. And like word embeddings, our goal is to map the entities (words or persona) to a vector space. Similarly analogous to word embedding where we map the semantics of the words, we map the traits of the speaker and addressee. By the way of which we aim to generate specialized or personalized responses.

We propose an end-to-end deep learning architecture to enhance multi-party dialogue generation. First, we introduce an environment-aware persona embedding to capture the changes in behaviour during the conversation. Second, we propose different techniques for adding this embedding to the existing baseline models, which include sequence-to-sequence, transformer and generative adversarial networks architectures. The most competitive framework is presented in Figure 1, which combines GANs with transformers.

A. Initialization of Persona Embedding

To generate such persona embedding, we consider that for each pair of dialogues there is a center utterance and a context utterance, where the center utterance is replying to the context utterance, and similarly the center speaker is replying to the context speaker. Hence, the probability distribution for a datapoint in our data set can be written as:

$$P(con_u \mid cen_{ub} \cap con_s)$$

where con_u and con_s stand for context utterance and context speaker respectively, and $cen_{ub} = cen_u \cap cen_s$ stands for center utterance-block.
Thus, our goal is to maximize

$$\prod_{cen_{ub}} \prod_{con_{ub}} P(con_u \mid cen_{ub} \cap con_s)$$

where $con_{ub} = con_u \cap con_s$ stands for context utterance-block, and con_u = context utterance, con_s = speaker of the context utterance. Note that we implicitly make $P(con_u \mid cen_{ub} \cap con_s)$ for non-existent or invalid pairs, close to 0.
We can re-write the above expression as follows:

$$max\ log \prod_{cen_{ub}} \prod_{con_{ub}} P(con_u \mid cen_{ub} \cap con_s)$$

Since $log(.)$ is a strictly increasing function, the minima and maxima won't be shifted.

$$max \sum_{cen_{ub}} \sum_{con_{ub}} log\ P(con_u \mid cen_{ub} \cap con_s)$$

which is equivalent to

$$-min \sum_{cen_{ub}} \sum_{con_{ub}} log\ P(con_u \mid cen_{ub} \cap con_s)$$

where $P(x)$ is the softmax function. Hence, through propagating this loss to our randomly initialized persona embedding matrix, we generate our desired persona embedding.

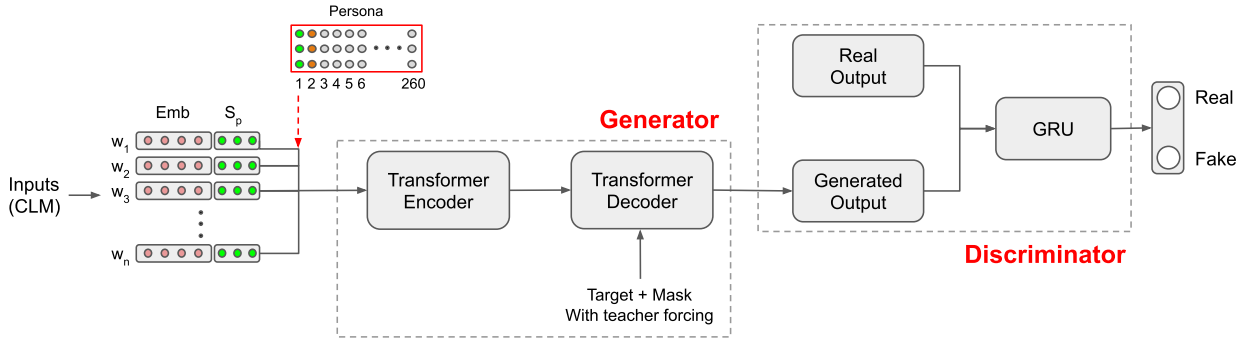

Fig. 1. Most competitive architecture combininf Transformers with GANs.

B. Context Representation using BiGRU

To encode an utterance, we first use Glove embeddings [18] to represent each word in a $\mathbb{R}^{300}$ vector space. We then initialize a random matrix ($\mathbb{R}^{100}$) as the persona embedding. Finally, we concatenate each word embedding with the persona embedding of its speaker to obtain a persona infused word embedding.

To learn the contextual relationship between words in an utterance, we pass the persona infused word embedding through a Bi-directional Gated Recurrent Unit ($BiGRU$) [5]. In particular, we deploy an attention mechanism as proposed in [2] to single out the words from the speaker which might have the most impact in generating the reply from the addressee, while also simultaneously capturing the persona of the speaker. We then use this context vector coupled with the attention outputs to generate an appropriate reply, i.e. generated response.

C. Transformer-based Generation

The persona-based contextual vector (i.e. representation of the utterance) is passed through a transformer [27] consisting of an encoder with self-attention and layer normalization. The obtained sentence encoding is then passed through a decoder consisting of a combination of self and multi-headed attention layers with layer norm. We use this combination of encoder and decoder as a means of generating a relevant contextual relationship between words of query utterance, and query and answer utterances. We choose the number of heads in the transformer carefully, so that to facilitate learning interpersonal relationship via the combination of persona and word embeddings.

D. Generative Adversarial Networks

Finally, the generation process is embedded into a GAN architecture. For the generator, we employ the transformer-based generation model as described above. The generator generates the reply for a person A when person B has spoken utterance U_B, while preserving the context of the conversation. Note that we use the cross entropy loss function to evaluate the losses in the generated dialogues. With respect to the discriminator, we use two distributions as the input, *viz.*(i). the generated output from the generator, and (ii). the real utterance from the data set. These inputs are then individually passed through a recurrent neural network and a multi-layered perceptron is used to determine which of the inputs comes from the real distribution or the inferred one. In particular, we use the binary cross entropy loss function, and the final back-propagated loss is the sum of losses from the generator and the discriminator.

IV. Datasets, Experiments, and Analysis

In this section, we present the details of the data sets that we used for our experiments, report and analyse experimental results, and compare with several state-of-the-art baseline systems.

A. Dataset

The Multi-Domain Wizard-of-Oz [1] data set is a large scale multi-turn conversational data set containing dialogues from numerous topics. MultiWoz has nearly 10K dialogues making it one of the largest dialogue generation data sets available. Combining the above two facts makes MultiWoz very viable to use as a data set to build an end-to-end dialogue system. We show the detailed description of the data set setups in the supplementary material accompanying this submission.

B. Experimental Setups

We use the Pytorch framework to implement our proposed models. We applied grid search to obtain the optimal parameters for each of the models presented. We show the description of hyper-parameters in the table IV obtained after model optimizations. Moreover, we use Adam as an optimizer and cross entropy as a loss function. For Transformers and GANs, we also used cosine annealing to prevent from overfitting. Models were then trained on GTX 2080 TI for 100 epochs.

C. Results and Analysis

To evaluate all the models, we use a wide-range of unsupervised metrics for language generation as proposed in [22], and not just a small part of them as in most related work. For each model, we also create two different variations: with persona

[1]https://tinyurl.com/yf875muj

Models	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr	STCS	EACS	VCS	GRMS
Transformer w/ persona	0.535	0.369	0.276	0.205	0.285	0.510	1.744	0.776	0.886	0.743	0.836
Transformer w/o persona	0.530	0.362	0.270	0.201	0.283	0.478	1.730	0.774	0.877	0.743	0.813
Seq2Seq w/ persona	0.350	0.151	0.120	0.092	0.201	0.330	0.492	0.770	0.871	0.713	0.835
Seq2Seq w/o persona	0.290	0.104	0.037	0.020	0.175	0.302	0.466	0.735	0.878	0.702	0.785
GAN w/ persona	**0.555**	0.398	**0.297**	**0.243**	0.291	**0.528**	**1.769**	**0.793**	**0.890**	**0.756**	**0.852**
GAN w/o persona	0.541	**0.407**	0.275	0.234	**0.295**	0.512	1.755	0.786	**0.890**	0.751	0.840

TABLE II

EVALUATION RESULTS USING VARIOUS UNSUPERVISED AUTOMATIC METRICS DISCUSSED IN [22], WHERE *Seq2Seq* IS THE MODEL USING ONLY BIGRU, *Transformer* IS THE MODEL USING ONLY TRANSFORMER NETWORKS WHILE GAN IS THE GENERATIVE ADVERSARIAL NETWORK ARCHITECTURE.

Statistics	***MultiWoz***
#Dialogue	8348
#Turns	113,556
#Tokens	1,490,615
Avg. turns per dialogue	13.46
Avg. tokens per turn	13.13
Total unique tokens	23,689

TABLE III

STATISTICS FOR THE *MultiWoz* DATASET.

Parameters	***Speaker Dependent***
Bi-GRU	200 *neurons*, *dropout=0.3*
Dense layer	100 *neurons*, *dropout=0.3*
Transformer	400 Neurons
Head	4
Restart	Cosine Annealing
Dense layer	100 *neurons*, *dropout=0.1*
Activations	*ReLu*
Optimizer	*Adam* (*lr*= 10^{-5})
Output	*Softmax*
Loss	*Cross-entropy*
Batch	*30*
Epochs	*100*

TABLE IV

MODEL CONFIGURATIONS.

embedding (w/ persona), and without persona embedding (w/o persona). A comparative study is shown in Table II.

We see that the GAN architecture outperforms every other existing models, and majority of the results can be attributed to the transformer-based architecture, which is a very powerful language generation model. We can also acknowledge the efficacy of our persona embedding, where we examine the improvements of results by adding or not the context-aware information. It is clear that in almost all configurations, the persona embedding plays a crucial role.

We also perform human evaluation to better understand the quality of the outputs produced from our proposed model. For the purpose of human evaluation, we had two annotators, and in case of any disambiguation in scores for data points (individual generated texts), we consider the rounded off average of the scores. The following characteristics were used for human evaluations:

- Fluency measures the grammatical correctness of the produced sentences,
- Adequacy measures the coherence of the generated response pertaining to the context,
- Persona Consistency measures the coherence of the generated dialogue in accordance with the social persona traits of the speaker.

Fluency, Adequacy and Persona Consistency are defined in the scale of 0-2, with '0' indicating wrong matching, '1' indicating acceptable matching and '2' indicating perfect matching. Results are illustrated in Table V. Interesting issues are evidenced as:

- GAN-based architectures seem to evidence more fluent utterances, while losing in adequacy over transformer-based only systems.
- GAN-based architectures made the best use of persona embeddings to generate person specific responses.
- Transformer-based architectures generate the most adequate responses with respect to the context of the conversation, while falling in other characteristics.
- *BiGRU*-based models is cleerly the least performing architecture.
- The addition of persona embedding improved the overall quality of generation in all of the characteristics for all of the models.

Models	F	A	PC
Transformer w/ persona	1.62	**1.18**	1.44
Transformer w/o persona	1.56	1.12	1.32
Seq2Seq w/ persona	1.16	1.06	1.28
Seq2Seq w/o persona	1.04	0.82	1.04
GAN w/ persona	**1.70**	1.10	**1.66**
GAN w/o persona	1.66	1.06	1.46

TABLE V

HUMAN EVALUATION FOR FLUENCY (F), ADEQUACY (A) AND PERSONA CONSISTENCY (PC) OVER 50 UTTERANCES BY TWO ANNOTATORS.

We observe the best scores in majority (2 out of 3 human evaluated metrics) for the GAN-based model, which is inline with our observation through the automated metrics (Table II).

D. Comparative Analysis

Models	***BLEU***
Baseline [1]	18.8
TokenMoE [17]	16.8
HDSA [3]	23.6
Structured Fusion [12]	16.3
LARL [29]	12.8
MarCo [26]	20.0
GAN w/ persona	**24.3**

TABLE VI

COMPARISON WITH STATE-OF-THE-ART MODELS.

We finally compare our approach against various systems in Table VI that made use of the same dataset and reported the results using only the BLEU-4 metric, which is the arithmetic mean of X-gram BLEU scores, for $X \in [1, 2, 3, 4]$. Evaluation clearly demonstrates the benefits of the architecture proposed in this paper. Moreover, it is important to note that:

- the state-of-the-art system presented in Table VI focused on policy optimization but had a secondary language generation task, whereas our sole task is language generation.
- the models presented in Table VI generated dialogues for only one of the two speakers in each conversation, while our model generated dialogues for both speakers, and hence the results for the same are used for comparison.

Hence, the addition of social persona traits while generating responses improved the performance of the GAN-based model to the extent that it outperformed the state-of-the-art model for the same dataset. We also saw from Table II that the inclusion of such embedding improved the quality of generation to beat all the models mentioned in Table VI except [3].

E. Detailed Analysis

	Dialogue
Good	I want to stay for 3 nights starting on thursday
	i would to book in 3 nights starting on tuesday
	I have a yellow audi booked for you the contact number is 07267474545 can i help you with anything else
	i have booked taxi bmw for for you the contact number is 07195096632 i help you with anything else
Bad	what is the address
	i is the price and
	that's ok i'll take the address if you have it
	i all i try the one and you can it for

TABLE VII
EXAMPLES OF "GOOD" AND "BAD" DIALOGUE GENERATIONS FROM THE GAN-BASED MODEL.

As per our analysis, we found out that our model with persona embedding focuses much more on generating responses starting with pronoun such as 'I'. This could be due to the heavy reliance of the proposed model on social persona trait while generating responses. Hence, in our observation the model tried to predict the *reaction* of the speaker on the given information (input). Hence, when the required generation did not require a personal sentence, i.e beginning with 'I', the generation started getting sidetracked significantly, (cf. *Bad* section in Table VII). Whereas, whenever such a personal sentence was required, our model generated fairly good sentences, (cf. *Good* section in Table VII). Moreover, to generate earlier dialogues in the conversation, the model had trouble in picking up the context correctly, which corrected itself over the duration of the conversation.

The proposed model was sufficiently good in capturing the context of the conversation as shown in examples presented in Table VIII. Here, we observe that the model performs well in both cases, i.e when generating a personalized sentence vs. generating a generic (non-personalized) sentence. On further analysis we also observed that the generated personalized sentences have a better context preservation with respect to the generated generic sentence.

	Dialogue
1	i have made a booking here is your information booking was successful reference number is dgghj53e
	i have booked those reservation for is your reference booking was successful reference number c73zmm0g 4ahco6oi is
2	for how many people would the reservation need
	yes how many people you reservation be the
3	i have found several trains that you can use what is your destination
	i can several 70 trains that meet can narrow what day your destination
4	okay do not hesitate to call us again
	thank have you hesitate to call if again if
5	sure where will you be departing
	i what are you be departing from
6	yes what is the entrance fee for that please
	i i is the entrance fee the museum
7	i'll be leaving from leicester on sunday
	i be leaving from cambridge and tuesday
8	thanks i'm also looking for a guesthouse in the east it doesn't need to have free wifi
	yes i also looking for a place in the north should need to have free parking
9	thank you for calling goodbye
	thank you for using have
10	i would like to travel on monday evening
	i have like to leave on tuesday

TABLE VIII
EXAMPLES OF PERSONALIZED DIALOGUE GENERATION FROM THE GAN-BASED ARCHITECTURE.

V. CONCLUSION

In this paper, we have successfully addressed the task of person-specific response generation in a multi-party setting. We have proposed a novel approach to learn a). the persona embedding of a speaker and b). the changes in the speaker's behaviour with resptec to other speakers involved in the conversation. In particular, our persona formalization allows to test many different generation architectures, from which transformer-based GAN frameworks outperform all other solutions for a wide range of automated and manual evaluations. Nevertheless, error analysis shows that strong improvements are still needed, especially in terms of fluency and adequacy to reach satisfactory results in real-world situations.

REFERENCES

[1] Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. Multiwoz - A large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5016–5026, 2018.

[2] Dushyant Singh Chauhan, Rohan Kumar, and Asif Ekbal. Attention based shared representation for multi-task stance detection and sentiment analysis. In *International Conference on Neural Information Processing (ICONIP)*, pages 661–669. Springer, 2019.

[3] Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. Semantically conditioned dialog response generation via hierarchical disentangled self-attention. In *57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3696–3709, 2019.

[4] Xiantao Chen, Jiaqi Mi, Menghua Jia, Yajuan Han, Moli Zhou, Tian Wu, and Daisong Guan. Chat with smart conversational agents: How to evaluate chat experience in smart home. In *21st International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI)*, pages 1–6, 2019.

[5] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *8th Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, 2014.

[6] Wenpeng Hu, Zhangming Chan, Bing Liu, Dongyan Zhao, Jinwen Ma, and Rui Yan. Gsn: A graph-structured network for multi-party dialogues. In *28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 5010–5016, 2019.

[7] Ran Le, Wenpeng Hu, Mingyue Shang, Zhenjun You, Lidong Bing, Dongyan Zhao, and Rui Yan. Who is speaking to whom? learning to identify utterance addressee in multi-party conversations. In *Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1909–1919, 2019.

[8] Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. SUMBT: Slot-utterance matching for universal and scalable belief tracking. In *57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5478–5483, 2019.

[9] Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. In *54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 994–1003, 2016.

[10] Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. Adversarial learning for neural dialogue generation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2157–2169, 2017.

[11] Qian Liu, Yihong Chen, Bei Chen, Jian-Guang Lou, Zixuan Chen, Bin Zhou, and Dongmei Zhang. You impress me: Dialogue generation via mutual persona perception. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1417–1427, 2020.

[12] Shikib Mehri, Tejas Srinivasan, and Maxine Eskenazi. Structured fusion networks for dialog. In *20th Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL)*, pages 165–177, 2019.

[13] Elnaz Nouri and Ehsan Hosseini-Asl. Toward scalable neural dialogue state tracking model. *arXiv preprint arXiv:1812.00899*, 2018.

[14] Oluwatobi Olabiyi, Alan Salimov, Anish Khazane, and Erik T. Mueller. Multi-turn dialogue response generation in an adversarial learning framework, 2019.

[15] Olabiyi Oluwatobi and Erik Mueller. DLGNet: A transformer-based model for dialogue response generation. In *2nd Workshop on Natural Language Processing for Conversational AI associated to ACL*, pages 54–62, 2020.

[16] Hiroki Ouchi and Yuta Tsuboi. Addressee and response selection for multi-party conversation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2133–2143, 2016.

[17] Jiahuan Pei, Pengjie Ren, and Maarten de Rijke. A modular task-oriented dialogue system using a neural mixture-of-experts. *arXiv preprint arXiv:1907.05346*, 2019.

[18] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[19] Alec Radford, Jeffrey Wu, Dario Amodei, Daniela Amodei, Jack Clark, Miles Brundage, and Ilya Sutskever. Better language models and their implications. *OpenAI Blog https://openai. com/blog/better-language-models*, 2019.

[20] Kiran Ramesh, Surya Ravishankaran, Abhishek Joshi, and K Chandrasekaran. A survey of design techniques for conversational agents. In *International Conference on Information, Communication and Computing Technology*, pages 336–350. Springer, 2017.

[21] Stephen Roller, Y-Lan Boureau, Jason Weston, Antoine Bordes, Emily Dinan, Angela Fan, David Gunning, Da Ju, Margaret Li, Spencer Poff, Pratik Ringshia, Kurt Shuster, Eric Michael Smith, Arthur Szlam, Jack Urbanek, and Mary Williamson. Open-domain conversational agents: Current progress, open problems, and future directions, 2020.

[22] Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *CoRR*, abs/1706.09799, 2017.

[23] Renuka Sindhgatta, Alistair Barros, and Alireza Nili. Modeling conversational agents for service systems. In Hervé Panetto, Christophe Debruyne, Martin Hepp, Dave Lewis, Claudio Agostino Ardagna, and Robert Meersman, editors, *On the Move to Meaningful Internet Systems: OTM 2019 Conferences*, pages 552–560. Springer International Publishing, 2019.

[24] Hui Su, Xiaoyu Shen, Pengwei Hu, Wenjie Li, and Yun Chen. Dialogue generation with GAN. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *32nd AAAI Conference on Artificial Intelligence (AAAI)*, pages 8163–8164, 2018.

[25] Aditya Nrusimha Vaidyam, Hannah Wisniewski, John David Halamka, Matcheri S Kashavan, and John Blake Torous. Chatbots and conversational agents in mental health: a review of the psychiatric landscape. *The Canadian Journal of Psychiatry*, 64(7):456–464, 2019.

[26] Kai Wang, Junfeng Tian, Rui Wang, Xiaojun Quan, and Jianxing Yu. Multi-domain dialogue acts and response co-generation. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 7125–7134, 2020.

[27] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Hugging face's transformers: State-of-the-art natural language processing, 2020.

[28] Rui Zhang, Honglak Lee, Lazaros Polymenakos, and Dragomir Radev. Addressee and response selection in multi-party conversations with speaker interaction rnns. In *32nd AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[29] Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1208–1218, 2019.