Cas-GANs: An Approach of Dialogue Policy Learning based on GAN and RL Techniques

Muhammad Nabeel[1], Adnan Riaz[2], Wang Zhenyu[3]
School of Software Engineering South China University of Technology, P.R. China[1, 3]
School of Computer Science and Technology, Dalian University of Technology, P.R. China[2]

***Abstract*—Dialogue management systems are commonly applied in daily life, such as online shopping, hotel booking, and driving booking. Efficient dialogue management policy helps systems to respond to the user in an effective way. Policy learning is a complex task to build a dialogue system. There are different approaches have been proposed in the last decade to build a goal-oriented dialogue agent to train the systems with an efficient policy. The Generative adversarial network (GAN) is used in the dialogue generation, in previous works to build dialogue agents by selecting the optimal policy learning. The efficient dialogue policy learning aims to improve the quality of fluency and diversity for generated dialogues. Reinforcement learning (RL) algorithms are used to optimize the policies because the sequence is discrete. In this study, we have proposed a new technique called Cascade Generative Adversarial Network (Cas-GAN) that is combination of the GAN and RL for dialog generation. The Cas-GAN can model the relations between the dialogues (sentences) by using Graph Convolutional Networks (GCN). The graph nodes are consisting of different high level and low-level nodes representing the vertices and edges of the graph. Then, we use the maximum log-likelihood (MLL) approach to train the parameters and choose the best nodes. The experimental results compared with the HRL, RL agents and we got state-of-the-art results.**

Keywords*—Generative Adversarial Networks (GANs); Graph Convolutional Network (GCN); Reinforce Learning (RL); Dialogue policy learning; Maximum Log-Likelihood (MLL)

I. Introduction

The task-oriented systems are used for interacting between user and the system using dialogues, that can be texts or spoken words. The response from the system is very important to know the good response of the trained system or agents. Efficient dialogue management policy help systems to respond to the user in an effective way. Meanwhile, end to end dialogue management agents are also known as dialogue agents or conversational agents. In dialogue management systems [1], the user interacts with the system by using the dialogues to train the task-oriented dialogue systems [2].

Typically, the experts design a dialogue management policy manually. The manual dialogue management policy is more time consuming, and it needs more cost to build the policy for the management system. Moreover, users want to get the output from the agent during the conversation. This research work relates the GAN with RL techniques to improve the dialogue policy learning in the dialogue management systems.

Reinforcement learning (RL) algorithms are used to optimize the policies because the sequence is discrete. The reinforcement learning mechanism is consisting of some states and corresponding actions. The generative adversarial networks (GANs) are used to train the dialogue policy learning, that is consisting of two networks, generator and the discriminator. The generator is responsible for generating the fake data and the discriminator is responsible for choosing the best reward from real and the generated data. The discriminator network issues a reward to inform the generator for generating a more realistic response. The discriminator network takes a dialogue consisting of a context-reply pair as input and outputs the probability which shows that the dialogues are coming from the real dialogues. There are many model-based evaluation methods used such as policy evaluation method. There are many model's free RL algorithms such as the Q-learning, SARSA, Dyna-Q and the temporal difference (TD).

The RL approach is more data-driven, because it trains the actions as the function of the states. Much research has been done to improve [3] the dialogue systems as well as the dialogue policy gradient [4]. The policy gradient is a policy method to train models that allow automatically generate the system response according to the trained model. The NN models play a key role to train such policy gradients, CNN, and RNN [5] to improve the training. Reinforcement learning (RL) algorithms are popularly used to solve the major problems in the task-oriented dialogue systems [6] to interact with the users. The general model of RL consists of three components are: (1) the state denoted by 's'; (2) the action denoted by 'a'; (3) the reward denoted is by 'r'. The policy is the rule that specifies how to choose an action. The action is depending on the awarded reward to verify the output either it is good or bad. Using a recurrent neural network (RNN) is making a significant improvement to learn the management policy [7] for helping the better system response. Recently, the GCN is used to learn the features through the functions in a graph. Also, GCN can be used to train the efficient management policy by working with the GAN.

The proposed approach (Cas-GAN) includes the general RL and GAN structure. The result shows the improvement when we trained the dialogue agent for operating dialogues using GANs platform using the proposed approach. The task-oriented systems are used for interacting between the user and the system. Users are able to interact with the system by using dialogues, and that can be texts or spoken words. The good response from the system is very important to know the quality of the trained system or agents. Efficient dialogue management policy help systems to respond to the user in an effective way.

The rest of this paper is structured as follows: related work in Section II. Section III introduces the methodology of the study. Section IV the proposed work. Section V explains the implementation. Results and analysis in Section VI. Finally, the conclusion and recommendation in Section VII.

II. Related Work

To train dialogues policy for user-system interaction (DMS), Satinder Singh and his team [8] have trained a spoken DMS that provides information to go for outing or do fun in New Jersey for users. The results show that the performance of the trained NJFun dialogue system was effectively improved. The NJFun dialogue system was implemented to give telephone access from the database of activities in New Jersey. Dialogue policy learning they define a state-based representation for dialogues. The results show the increasing number of completed dialogues from 46 % to 69 % during the training and that considered as the significant improvements for several reward measures.

A statistical model to perform real-time policy learning in the spoken dialogue management systems and updating of dialogue states has been proposed by Blaise Thomson [9]. This framework is based on POMDP it provides a well-established statistical model of the spoken DMS. A spoken DMS was proposed to allow users for better decision making. Which bridged the gap between the user's criteria, and if not found responded with an alternate [10].

Another proposed model of dialogue state considers the user preferences as well as the user knowledge about the domain transferring from restaurant booking to car booking. The result shows that the learned policy works better than several baseline methods. The recent work used NN models [11] for generating dialogues, and that shows much improvement and efficient responses for conversational agents. But modeling the future responses regarding dialogues are very crucial. Also, that demands the use of traditional NLP models of dialogue with the help of RL. The proposed policy gradient method works better because it initializes the encoder, decoder, and RNN using the MLE parameters.

The DMS builds a dialogue agent that can fulfill the desired complex tasks is much challenging because of different multiple subtasks such as travel planning. The dialogue manager is consisting of a high-level dialogue policy able to selects between subtasks. The low-level policy selects the required actions to complete the subtasks received from the high-level policy. The end-to-end dialogue model architecture [12] consists of the user utterances, the LSTM dialogue state, also the slot values, policy network, and the system action. They apply the REINFORCE algorithm to optimize the network parameters and used soft-max policy during RL training to encourage the agent to explore the dialogue action space.

The GANs has been recently applied to Neural Machine Translation system (NMT). Zhen Yang builds a conditional sequences GAN [13] consisting of two adversarial sub-models the generator and discriminator. Each one was able to respond and generate sentences that were hard to discriminate from human translated sentences. Discriminator was also used to discriminate the machine-generated sentences from human translated sentences. The GANs sentence level BLEU is utilized as the reinforce objective for the generator, and biases the generation towards high BLEU points. The proposed model presents a divide and conquers method that discovers the hidden structure of the tasks to enable effective policy learning.

Da Tang proposed Sub-goal Discovery Network [14] to divide the complex goal-oriented tasks into sub-goals in an unsupervised way. Moreover, they present a dialogue agent for the composite task of travel planning. There are two processes used to train the dialogue agent. One is a high-level process that selects the sub-goals to complete, and second is a low-level process that chooses the primitive actions to accomplish the selected objective sub-goals. The experiment results show the learned agent performs efficiently against an agent learned using expert-defined sub-goals. The sub-goal discovery trained by using RNN model has been carried out by two RNN models used in this approach, RNN1, and RNN2. RNN2 provides information about previous states from RNN1. To train SDN, RMSProp has been used to optimize model parameters. The result shows better performance as compared to the state-of-art baseline models.

III. Methodology

We have proposed a new approach for dialogue policy learning and introduced the Cascade Generative adversarial network (Cas-GAN). In this approach, the generator part is responsible for the training of the policy by using GCN and RL techniques. The generator is responsible for generating the fake data and the MLL technique is used to calculate the generator value [15]. The discriminator is responsible to learn the strategies by using the MCS and it will generate a reward which is depending on the value from 0 to 1. The reward values 0 and 1 are representing the bad and good reward. This approach will lead to efficient dialogue policy learning to allow users to find desired search results more accurately in a short time by interacting with the RL [16] based on the generated system.

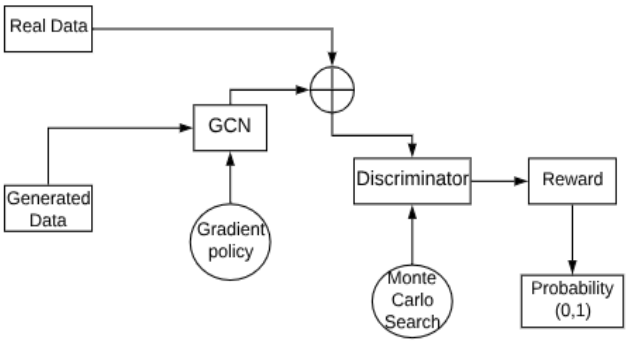
IV. Proposed Work

To train the task-oriented dialogue systems the dialogue policy plays a vital role. If dialogue policy is learned efficiently that lead to effective results in the responding system [17]. Typically, the experts design a dialogue management policy by hand that is more time consuming and needs more cost. Recent research has suggested that efficiently learned policy using formalisms of RL and the MDP. RL algorithms are popularly used to solve the major problems in task-oriented dialogue

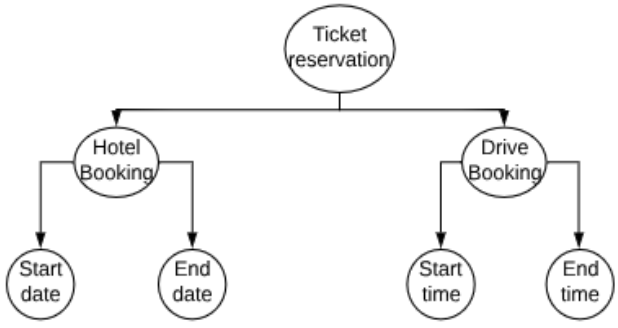
systems. The general model of RL is consisting of three components are (1) the state denoted by s; (2) the action denoted by a; (3) reward denoted by r. The policy is the rule that specifies to choose an action the level of action good or bad is depending on the awarded reward. The architecture in Fig. 1 shows the generator has been trained with fake data. Moreover, policy learning is also learned within the generator. The policy has been learned by GCN and RL techniques, and we have followed the general architecture of RL that is consisting of some states and actions. Depending on these actions the reward will be issued to the system to checking user satisfactory to reach the goal, for example, booking air ticket and hotel reservation or taxi drive booking. The discriminator is responsible for choosing the best reward for the system response, and there are several methods have been used to train the discriminator. We have used the reinforce algorithm along with MCS for choosing the best reward.

The proposed technique shows that we have used GCN to learn the gradient policy for better system response. The discriminator will use MC search method to choose the best reward according to the user search query.

The dialogue sequence generation is a common problem to overcome this challenge, we have used structured dialogue sequences dataset to train a parameterized θ generative model Gθ for generating the sequence G1: T = (g1..., gt, T), the capital G denotes the vocabulary for the candidate tokens. We have considered a policy learning problem in RL based scenario. In time step t, the states are the currently produced tokens (g1..., gt−1). Whereas, the action 'a' is the next token yt to be selected. The policy model is stochastic that is denoted by Gθ (gt|G1: g−1).



(a) Architecture of Cas-Gan.



(b) Description of Graph Nodes.

Fig. 1. Visualization of Cas-GAN.

We have trained a φ-parameterized discriminative model Dφ, to provide the guidance for improving generator Gθ for dialogue generation [18]. Where Dφ (G1: T) is the probability indicating the likelihood of a required dialogue G1: T from real data or the generated data. The discriminative Dφ has been trained by giving positive utterances from the true sequence dialogue data, and negative examples from the synthetic sequence dialogues that are generated by the generative model Gθ. On the other hand, generative Gθ has been updated by learning a policy gradient also MCS is used for the expected rewards to be received from the discriminative model Dφ. The reward is calculated by the likelihood estimation and that shows how successful it can fool the discriminative model Dφ. The problem of classifying nodes in a graph can be denoted as graph-based semi-supervised learning shows the information of labeling the data is very smoothed over the graph via some form of explicit graph-based regularization [19]. We have used a graph Laplacian regularization. The loss function can be defined as:

$$\mathcal{L} = \mathcal{L}_0 + \lambda \mathcal{L}_{reg},\ with\ \ \mathcal{L}_{reg} = \sum_{i,j} A_{ij} ||f(X_i) - f(X_j)||^2 = f(X)^T \Delta f(X) \quad (1)$$

In Eq (1), L_0 denotes the supervised loss w.r.t the labeled part of the graph, and f (·) is a NN differentiable function. However, the λ is a weighting factor, and X is a matrix of node feature vectors, Xi = M- N denotes the unnormalized graph Laplacian of an undirected graph G = (V, E) with the nodes N∈ V, edges (vi, vj) ∈ E, adjacency matrix A ∈ R, N×N (binary or weighted) and degree matrix M_ii = A_ij. The formulation is based on the supposition to connect the nodes in the graph that are likely to share the same label. The assumption may limit the modeling capacity that the graph edges do not need necessarily encode node similarity but it may contain the additional information.

The objective of the dialogue policy learning Gθ (yf|Y1: f−1), to train or develop a dialogue sequence from the starting state s0 for maximizing the expected reward. Eq (2) defines the form to complete a sequence of dialogue utterance.

$$J(\theta) = \mathbb{E}[R_T | s_0, \theta = \sum_{y1 \epsilon Y} G_\theta(y1|s_0).\, Q_{D_\theta}^{G_\theta}(s_0, y1) \quad (2)$$

In Eq (2), R_T is the reward for a complete sequence of dialogue utterance and values of reward comes from the discriminator. Q_(D_θ)^(G_θ) (s,a) is the action-value function of a sequence which is consists of a generator and the discriminator of the GAN network. The expected accumulative reward is starting from the state (s) taking action (a) and followed by the policy Gθ. The rationale of the objective function for a sequence is starting from a given initial state. The goal of the generator is to generate a dialogue sequence such as (i.e. If the user wants ticket reservation for different countries or the user wants to book the hotel at the same time the user can also select to reserve the car driving. In all this scenario the system should give an efficient response and that is following by the discriminator reward to consider if it comes from the truth data [20]. The question is how we can estimate the action-value function. To overcome this problem, we used the REINFORCE algorithm that will consider the estimated probability of being real from the discriminator Dφ (Y n1: T) as the reward value.

$$\mathrm{Q}_{\mathrm{D}_\theta}^{\mathrm{G}_\theta}(\mathrm{a}=\mathrm{y}_\mathrm{T}, \mathrm{s}=\mathrm{Y}_{1:\mathrm{T}-1}) = \mathrm{D}_\emptyset(\mathrm{Y}_{1:\mathrm{T}}) \quad (3)$$

Although we can get the reward value from the discriminator for finished dialogue sequence utterance we further focus on the long-term reward. At every time step, we don't focus only on the fitness of previous tokens but also on the future resulted in the outcome. It is the same to play the Go or Chess games show the players can end up the immediate interests sometimes for the long-term win. For the evaluation of value-based rewards to a corresponding state, we used MCS with a roll-out policy Gβ, to sample the unknown last T −t tokens [21].

$$\{\mathrm{Y}_{1:\mathrm{T}}^1, \ldots, \mathrm{Y}_{1:\mathrm{T}}^\mathrm{N}\} = \mathrm{MC}^{\mathrm{G}_\beta}(\mathrm{Y}_{1:\mathrm{f}}; \mathrm{N}) \quad (4)$$

In Eq (4), ⟦ Y⟧ _m1: t = (y1... yt) and Y n t+1: T is depending on the roll-out policy Gβ, and the current state. In this work, Gβ is set the same as the generator. To overcome the variance and find the efficient result from the action-value we trained the roll-out policy starting from the current state until the end of the dialogue sequence utterance for N times batch samples of the output.

$$Q_{D_\theta}^{G_\theta}(s = Y_{1:f-1}, a = yt) = \{D_\emptyset^{\frac{1}{M}} \sum_{m=1_{(Y1:f)}}^{N} D_\emptyset(Y_{1:T}^m \in MC^{G_\beta}(Y_{1:f};\ M) \quad \begin{matrix} for & f < T \\ for & f = T \end{matrix} \quad (5)$$

In Eq (5), there is no immediate reward. The function is iterative, and the next state value is starting from the state's' = Y1: f, and rolling out to the end. The purpose of using the discriminator Dφ as a reward function can be dynamically updated to the required improvements of the generative model. Thus, once we get a set of more true generated dialogue utterances and we should train the discriminator model again from the Eq (6).

$$\min_\emptyset -\mathbb{E}_{Y\sim p_{data}}[\log D_\emptyset(Y)] - \mathbb{E}_{Y\sim G_\theta}[\log(1 - D_\emptyset(Y))] \quad (6)$$

When we obtain a new discriminator model D_∅, we are also ready for updating the generator G_θ for next utterance. The method of gradient policy learning depends on optimizing a parametrized policy to directly maximize the long-term reward. The gradient of the objective function J(θ) w.r.t. the generator's parameters θ.

$$\nabla_\theta J(\theta)\mathbb{E}_{Y_{1:f-1\sim G_\theta}}[\sum_{yf\in Y} \nabla_\theta\, G_\theta(yt|Y_{1:f-1}).\, Q_{D_\emptyset}^{G_\theta}(Y_{Y1:f-1,yf})] \quad (7)$$

Eq (7) is representing the deterministic state transition and the zero intermediate rewards. By using the likelihood ratios, we have built an unbiased estimation shown in Eq (8, 9).

$$\nabla_\theta J(\theta) \simeq \frac{1}{T}\sum_{f-1}^{T}\sum_{yf\in Y} \nabla_\theta\, G_\emptyset(yf|Y_{1:f-1}).\, Q_{D_\emptyset}^{G_\theta}(Y_{1:f-1,yf})$$

$$= \frac{1}{T}\sum_{f=1}^{T}\sum_{yf\in Y} \mathrm{G}_\theta\,(yf|Y_{1:f-1})$$

$$\nabla_\theta log G_\theta\,(yf|Y_{1:f-1}).\, Q_{D_\emptyset}^{G_\theta}(Y_{1:f-1,yf}) \quad (8)$$

$$= \frac{1}{T}\sum_{f=1}^{T} \mathbb{E}_{yf\sim G_\theta(yf|Y_{1:f-1})}$$

$$[\nabla_\theta log G_\theta\,(yf|Y_{1:f-1}).\, Q_{D_\emptyset}^{G_\theta}(Y_{1:f-1,yf})] \quad (9)$$

Y1: f−1 is the learned state that is sampled by the Gθ. Moreover, the expectation E [·] can be approximated by sampling the methods and we can update the generator's parameters in Eq (10).

$$\theta \leftarrow \theta + \alpha_h\, \nabla_\theta\, J(\theta) \quad (10)$$

The α_ (h)∈ R+ indicates the learning rate, corresponding at the h-th step. We can also use advanced gradient algorithms, such as Adam and RMSProp.

V. Implementation

In order to test the efficiency and the understanding of Cas-GAN, we have pre-trained the data in the generator network by using the dataset of dialogue sequences. For simulating the real-world structural dialogues, we used a model of language to indicate the dependency for the free slots by issuing tokens.

In order to set the experiment, we first initialize the parameters of a GAN network, followed by the normal distribution between different states denoted by N (0,1). The normal distribution helps to distribute the data by using MLE [16]. The generated data along with the real data are being concatenated under the mechanism of GAN. We used G oracle to generate dialogues of length 15 to 20, for the training set S to the generative models. The occurrences are depending on the number of turns, also the system and user takes to finish one whole dialogue in different frames. In the training set for the discriminator, the reward is representing from 0 to 1 that indicates the quality of the response sent to the system. By follows the reward, the generator is learning more efficiently to response the user queries. And try to choose the best matching dialogue sent to the discriminator using Monte Carlo search [22]. For different tasks to be performed, we must design some specific structures for the convolutional layer. In this experiment, the kernel size is from 1 to T and the number of each kernel size is between 100 to 2004. The dropout and L2 regularization are used to avoid over-fitting.

The learning process of the new technique shows better performance when we are comparing the results with the previous technique HRL, RL and Rule. In the training process, the number of turns is changing from the frames to get the true previous tokens, also from one state to another followed by the desired query of the user. Curriculum rate ω is used to overcome the problem for the probability by replacing the true tokens with the generated ones. To get stable and effective performance, we decreased ω by 0.002 for every training epoch that gives a good probability rate. We have used BLEU [23] as standard to measures the similarity from the real and generated reward for scoring the finalize samples from Monte Carlo Search.

The experimental dataset for this experiment is used to train the policy for the GANs, by using RL platform. The publicly available multi-domain dialogue corpus called Q/A frames dataset [24] have been used in this experiment. The dataset

provides the basic knowledge for generating the data, also the discriminator can distinguish between the real and the generated data by the generator. We used the epochs size 200 for the adversarial pre-training and 115 for pre-training the generator. There is 1369 total number of dialogues that have divided into different frames, and 268 hotels from 109 cities. The average user satisfaction rate is 4.58 regarding bad, average, good, very good, and excellent that is representing from 1 to 5 respectively. The number of turns for the whole dataset is 19986.

This dataset includes three types of dialogues are (1) flight reservation; (2) hotel reservation; (3) the driving reservation. This dataset is prepared by domain experts and publicly available on the Maluuba website. There are different representations of the dataset that are including (1) the number of occurrences according to action names; (2) turns per number of actions; (3) dialogue act frequency according to the number of dialogue act frequency; (4) the number of frames and ratio of frames changes; (5) dialogue length distribution according to number of turns for per dialogue length. User and system or wizard occurrences also the representation for package comparisons of frames and the linear frames comparison.

The other dialogue corpus dataset is also available, and most of them are consisting of the datasets of movie booking or the restaurant booking. For all these datasets the interaction between the user and the system is important because these datasets purely prepared for the chatbot systems. The used dataset provides more information about the travel booking, hotel booking, and drive booking. The representation of the data is showing the distribution of the dataset, and the distribution of the frame information. The most commonly used occurrence used is 'inform' when the user or system is interacting with each other using some dialogues. The first action performed is to inform the state and performed a suitable action according to the user information.

VI. Results and Analysis

The epoch size is defining as 200 for adversarial pre-training, and 115 for pre-training the generator. There is 1369 total number of dialogues divided into different frames, and 268 hotels from 109 cities. In pre-training, the generator the epoch range is 0 to 115. The test loss starts from 10.169 which can be accumulated as 1 then it decreases up to 115 epochs. The test loss is around 9.09 can be denoted as 0.9 according to 0 - 1 value. In the adversarial training, the training value starts from 9.085 after accumulates it become 0.985 as per turns. We used 200 epochs for the discriminator part, at last, we get the test loss as 8.77 and can be accumulated as 0.877 after the evaluation as per setting value from 0 to 1. These values are comparatively better than the baseline models for this work.

Table I shows the results of our proposed approach. We have shown the success rate and number of turns, also reward values have performed efficiently better than the previous techniques. The result of the new technique has compared with the previous techniques as shown.

TABLE. I. Performance of Agents

Agent names	Success rate of Agent	Number of Turns	Number of Rewards
Rule	.3210	46.21	-24.08
RL	.4432	45.30	-1.835
HRL	.6419	44.26	35.38
Cas-GAN	**.6625**	**43.10**	**35.70**

TABLE. II. Performance of Model Training

Agent	**Number of epochs for pre-training**	**Number of epochs for adversarial**	**Adversarial loss**	**Pre-training loss**
Seq-GAN	115	150	8.71	9.24
Cas-GAN	**115**	**200**	**8.67**	**9.08**
RL	115	150	9.45	10.17

Table II shows that the pre-training loss for the previous technique (SeqGAN) is more than our technique (Cas-GAN). The pre-training loss for 115 epochs is 9.21, and for adversarial training, it is 8.73. While for CasGANs pre-training loss is less than the previous technique (Seq-GAN) as 8.67 for 200 epochs and 9.08 for 115 pretraining epochs. While for the RL agent the pretraining loss is 10.17 and the adversarial loss is 9.45 is higher than our approach. It shows the model works more efficient when to train the dialogues and it gives more related information about the user query. The test loss is considered as the unmatched number of dialogues in a conversation.

VII. Conclusion

Recent research has suggested dialogue policies can be efficiently learned by using formalisms of reinforcement learning and Markov decision process. This work introduced a new approach to learn the dialogue policy for the efficiency of user-system responses. Cas-GANs is using the GANs network architecture and connecting to the GCN for generating the fake data by the generator network of GANs. The discriminator part successfully chooses the best rewards for the user query and sent it back to the generator for the next queries. The proposed technique is a new addition to GANs and it is comparable to the previous techniques HRL, Rule etc. To train the policy learning by decreasing the number of turns in the shape of dialogue terms shows that the model is working better and more efficiently trained by improving the policy-based RL techniques. In future, we can improve the modeling techniques for the improvement of dialogue policy learning, and can apply this proposed model to the other datasets.

REFERENCES

[1] Serban, Iulian V., Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. "Building end-to-end dialogue systems using generative hierarchical neural network models." AAAI'16 Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence", 2016: 3776-3783.

[2] Sordoni, Alessandro, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. "A neural network approach to context-sensitive generation of conversational responses." In Proc. of NAACL-HLT", 2015: 196-205.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. "VQA: Visual Question Answering". In Proceedings of the IEEE International Conference on Computer Vision, 2015: 2425–2433.

[4] Liu, Chia-Wei, Ryan Lowe, Iulian V. Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. "How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation." Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016: 2122– 2132.

[5] Zeiler, Matthew D and Fergus, Rob. "Visualizing and understanding convolutional networks". In Computer Vision–ECCV 2014, 2014: 818–833.

[6] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, Graves A, Riedmiller M, Fidjeland AK, Ostrovski G, Petersen S. "Human-level control through deep reinforcement learning." Nature. 2015 18(7540): 529–533.

[7] Jen-Tzung Chien and Yuan-Chu Ku. "Bayesian recurrent neural network for language modeling". IEEE transactions on neural networks and learning systems 2016, 27(2):361–374.

[8] Singh, Satinder, Diane Litman, Michael Kearns, and Marilyn Walker. "Optimizing dialogue management with reinforcement learning: Experiments with the NJFun system." Journal of Artificial Intelligence Research, 2002, 16 (1): 105-133.

[9] Thomson, Blaise, and Steve Young. "Bayesian update of dialogue state: A POMDP framework for spoken dialogue systems." Computer Speech & Language, 2010, 24 (4): 562-588.

[10] Misu, Teruhisa, Komei Sugiura, Kiyonori Ohtake, Chiori Hori, Hideki Kashioka, Hisashi Kawai, and Satoshi Nakamura. "Modeling spoken decision making dialogue and optimization of its dialogue strategy." In Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2010: 221-224.

[11] Ji He, Jianshu Chen, Xiaodong He, Jianfeng Gao, Lihong Li, Li Deng, and Mari Ostendorf. "Deep reinforcement learning with a natural language action space". In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016: 1621–1630.

[12] Sainbayar Sukhbaatar, arthur szlam, Jason Weston, and Rob Fergus. "End to-end memory networks". In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, Advances in Neural Information Processing Systems 28, 2015, 2440–2448.

[13] Yang, Zhen, Wei Chen, Feng Wang, and Bo Xu. "Improving neural machine translation with conditional sequence generative adversarial nets." Proceedings of NAACL-HLT 2018, 2017: 1346–1355.

[14] Tang, Da, Xiujun Li, Jianfeng Gao, Chong Wang, Lihong Li, and Tony Jebara. "Subgoal discovery for hierarchical dialogue policy learning." Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018: 2298–2309.

[15] Jewell NP, Kalbfleisch JD. "Maximum likelihood estimation of ordered multinomial parameters". Biostatistics, 2004, 5(2): 291-306.

[16] Liu, Bing, Gokhan Tur, Dilek Hakkani-Tur, Pararth Shah, and Larry Heck. "End-to-end optimization of task-oriented dialogue model with deep reinforcement learning." arXiv preprint arXiv:1711.10712, 2017.

[17] Li, Xiujun, Yun-Nung Chen, Lihong Li, Jianfeng Gao, and Asli Celikyilmaz. "End-to-end task-completion neural dialogue systems." Proceedings of the The 8th International Joint Conference on Natural Language Processing, 2017: 733–743.

[18] Hamilton W, Ying Z, Leskovec J. "Inductive representation learning on large graphs". In Advances in Neural Information Processing Systems, 2017: 1024-1034.

[19] Abruna, Joan, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. "Spectral networks and locally connected networks on graphs." In International Conference on Learning Representations (ICLR2014), 2014.

[20] Browne, Cameron B., Edward Powley, Daniel Whitehouse, Simon M. Lucas, Peter I. Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. "A survey of monte carlo tree search methods." IEEE Transactions on Computational Intelligence and AI in games, 2012, 4(1): 1-43.

[21] Liu, Bing, and Ian Lane. "An end-to-end trainable neural network model with belief tracking for task-oriented dialog." DOI: 10.21437/Interspeech, 2017.

[22] Lowe, Ryan, Iulian V. Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. "On the evaluation of dialogue systems with next utterance classification." Association for Computational Linguistics, 2016: 264–269.

[23] Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. "BLEU: a method for automatic evaluation of machine translation." In Proceedings of the 40th annual meeting on association for computational linguistics, 2002: 311-318.

[24] Schulz H, Zumer J, Asri LE, Sharma S. "A frame tracking model for memory-enhanced dialogue systems". Proceedings of the 2nd Workshop on Representation Learning for NLP, 2017: 219–227.