Self-Supervised Dialogue Learning

Jiawei Wu and **Xin Wang** and **William Yang Wang**
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, CA 93106 USA
{jiawei_wu,xwang,william}@cs.ucsb.edu

Abstract

The sequential order of utterances is often meaningful in coherent dialogues, and the order changes of utterances could lead to low-quality and incoherent conversations. We consider the order information as a crucial supervised signal for dialogue learning, which, however, has been neglected by many previous dialogue systems. Therefore, in this paper, we introduce a self-supervised learning task, *inconsistent order detection*, to explicitly capture the flow of conversation in dialogues. Given a sampled utterance pair triple, the task is to predict whether it is ordered or misordered. Then we propose a sampling-based self-supervised network $\mathcal{SSN}$ to perform the prediction with sampled triple references from previous dialogue history. Furthermore, we design a joint learning framework where $\mathcal{SSN}$ can guide the dialogue systems towards more coherent and relevant dialogue learning through adversarial training. We demonstrate that the proposed methods can be applied to both open-domain and task-oriented dialogue scenarios, and achieve the new state-of-the-art performance on the OpenSubtitles and Movie-Ticket Booking datasets.

1 Introduction

In recent years, dialogue systems have achieved fruitful results with neural conversation models in both open-domain generation (Ritter et al., 2011; Sordoni et al., 2015b; Li et al., 2016b, 2017; Xu et al., 2017; Zhang et al., 2018b) and task-oriented completion (Wen et al., 2015, 2017; Williams et al., 2017; Bordes et al., 2017; Su et al., 2018). These methods empower lots of real-world dialogue applications such as Google Home and Apple Siri.

However, the utterance generation from dialogue systems still faces some critical challenges, including utterance blandness and incoherence (Gao et al., 2018). They are mainly caused by the objective function of the dialogue systems that prefer utterances with unconditionally high probability (Li et al., 2016a). We argue that in a meaningful and coherent dialogue, the change of utterance order will lead to a low-quality dialogue. However, most existing neural-based dialogue systems either encode the full dialogue history (Li et al., 2017; Xu et al., 2017) or only the current utterance (Liu and Lane, 2018). None of them explicitly models the sequential order and studies its criticality to the dialogue learning problem.

In this paper, we explore the sequential order within the dialogue as the self-supervised signal to guide meaningful and coherent dialogue learning. We introduce a self-supervised learning task, inconsistent order detection, to explicitly capture the order signal of the dialogue. The task is defined as, given a target utterance pair triple, the model is required to predict whether the triple is correctly ordered or not. For instance, the utterance pair triple $\langle (Q_1, A_1), (Q_4, A_4), (Q_2, A_2) \rangle$ is misordered. The key to solving this task is to model the utterance order based on the dialogue context effectively. But when directly encoding the full dialogue history along the temporal order, the model actually only focuses on the ending utterances, and earlier information is largely discarded (Li et al., 2017). Thus, we propose a sampling-based **self-supervised network** ($\mathcal{SSN}$) to account for the forgetfulness problem and solve the inconsistent order detection task. In order to accurately predict if a target utterance triple is ordered or not, we randomly sample utterance triples from the dialogue history as the reference to incorporate the dialogue context. Since for the same target utterance triple, the sampled triple references are different at different iterations during training. It essentially approximates the full dialogue history without suf-

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3857–3867
Florence, Italy, July 28 - August 2, 2019. ©2019 Association for Computational Linguistics

fering from the forgetfulness issue.

To further utilize $\mathcal{SSN}$ in real dialogue learning, we propose to jointly learn $\mathcal{SSN}$ and the dialogue model via alternative training, where the output probability of $\mathcal{SSN}$ is treated as the order signal to evaluate the generated utterance. Moreover, the proposed approach can be applied to both open-domain and task-oriented dialogue learning, which indicates that $\mathcal{SSN}$ is a general and scalable approach for dialogue learning. Empirical results on two widely-used benchmark datasets, OpenSubtitles and Movie-Ticket Booking, show that our self-supervised network consistently improves the state-of-the-art (SOTA) neural-based dialogue training methods. In summary, our main contributions are three-fold:

- We introduce the task of inconsistent order detection, and propose a self-supervised learning network $\mathcal{SSN}$ to solve this task and explicitly model the crucial order information in dialogue.
- We propose a general framework to jointly learn $\mathcal{SSN}$ and the dialogue models, where the sequential order in dialogues can be explicitly used to guide the utterance generation.
- Our method advances the existing state-of-the-art dialogue systems in both open-domain and task-oriented scenarios.

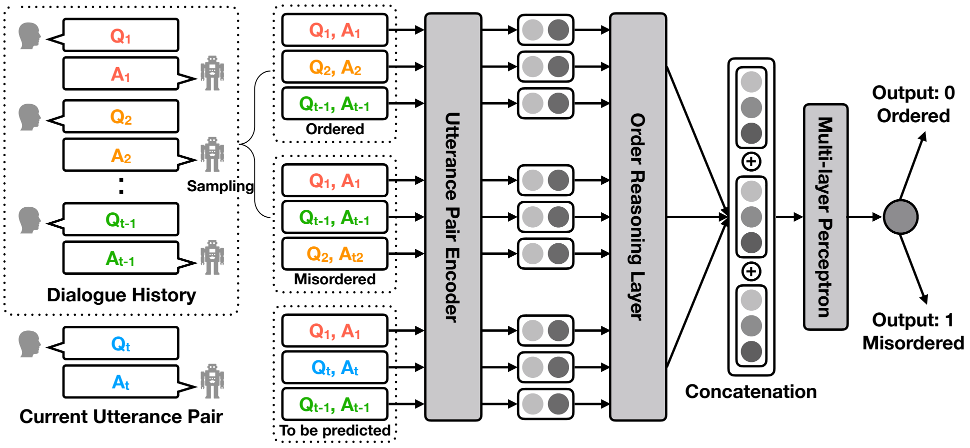
2 Related Work

Dialogue Learning Dialogue systems can be roughly classified into open-domain and task-oriented scenarios. In recent years, neural-based conversation models have shown great power in building dialogue systems (Ritter et al., 2011; Sordoni et al., 2015b; Vinyals and Le, 2015; Serban et al., 2016; Luan et al., 2016). However, the utterances generated by neural-based dialogue systems still suffer from blandness and incoherence (Gao et al., 2018). To address these problems, Li et al. (2016a) propose a mutual information objective to infer the utterance generation. Serban et al. (2017) and Zhang et al. (2018a) further apply the latent variable models to generate more specific responses. Similar to some language generation tasks (Lamb et al., 2016; Yu et al., 2017), Generative adversarial networks (GAN) (Goodfellow et al., 2014) have also been adapted to learn a better objective function for the dialogue (Li et al., 2017; Xu et al., 2017; Liu and Lane, 2018; Su et al., 2018). The discriminator in GAN is often used to evaluate the generated utterances and guide dialogue learning. However, these methods mainly focus on the surface information of generated utterances to guide the dialogue learning, and fail to consider the utterance connection within the dialogue history. In this paper, we focus on the sequential information of the dialogue and show that the unique sequential order in a meaningful and coherent dialogue contains more useful semantic information for dialogue learning.

Self-Supervised Learning Self-supervised learning, which aims to train a network on an auxiliary task where ground-truth is obtained automatically, has been successfully applied in computer vision. Many self-supervised tasks have been introduced to use non-visual but intrinsically correlated features to guide the visual feature learning (Doersch et al., 2015; Wang and Gupta, 2015; Pathak et al., 2016). As for natural language processing, predicting nearby words (Mikolov et al., 2013b,a) is a self-supervised task to learn word embeddings. The language modeling is another line of self-supervision where a language model learns to predict the next word given the previous sequence (Bengio et al., 2003; Dai and Le, 2015; Peters et al., 2018). Recently, Devlin et al. (2019) further proposes two self-supervised tasks, the masked language model and next sentence prediction, to learn sentence embeddings. Lample and Conneau (2019); Liu et al. (2019) further extend these two tasks into multi-lingual and multi-task paradigms. Wang et al. (2019) consider them at the sentence-level for extractive summarization. Our work is the first to consider the sequential order as the self-supervised signal in dialogue and we propose the self-supervised task of inconsistent order detection towards more coherent and relevant dialogue learning.

3 Methods

In this section, we systematically describe how to utilize the internal sequential order of utterances as self-supervision for dialogue learning. In Section 3.1, we first introduce the task of inconsistent order detection, where the model needs to predict whether one sampled triple of the dialogue is correctly ordered or not. We then present an effective sampling-based approach, self-supervised network ($\mathcal{SSN}$), to learn to capture the important



(a) Triple Reference Sampling **(b) Inconsistent Order Prediction**

Figure 1: The overview of our self-supervised network ($\mathcal{SSN}$) for inconsistent order detection. Given a target triple containing the current utterance pair (Q_t, A_t) to be predicted, (a) we first sample triple references from previous dialogue history $\{(Q_1, A_1), \cdots, (Q_{t-1}, A_{t-1})\}$ in each iteration. The references can be ordered or misordered. (b) For each triple, it is transformed into the triple embedding. The concatenation of triple embeddings is fed into a MLP, and gives the probability based on the current sampling.

order signal and solve this task (see Section 3.2). In the end, we show in Section 3.3 how $\mathcal{SSN}$ can contribute to both open-domain and task-oriented dialogue learning by modeling the inconsistent order detection.

3.1 Inconsistent Order Detection

The dialogue systems aim at conversing with the human in a meaningful and coherent way (Gao et al., 2018). Thus, the sequential order in dialogue data is an important signal for building a good dialogue system. Existing neural-based dialogue systems only consider this signal in a weak and implicit way, where they use hierarchical encoders to model the dialogue history (Sordoni et al., 2015a; Serban et al., 2016; Li et al., 2017; Serban et al., 2017; Xing et al., 2018). However, we argue that these methods are mainly designed to model the overall semantic context information of the dialogue history but not good at modeling intermediate sequential order. Especially, the order signal is becoming weak as the number of dialogue turns increases. Thus, we propose the task of inconsistent order detection to force building models to capture this signal as self-supervision explicitly. Given a dialogue till the turn t, we can formulate it as $\{(Q_1, A_1), (Q_2, A_2), \cdots, (Q_t, A_t)\}$, where (Q_t, A_t) is a pair of human-machine utterances. Then we can sample multiple triples of this dialogue as utterance pair triples using the following strategies:

- **Ordered triple sampling**: We sample a triple following the dialogue sequential order as $\langle (Q_i, A_i), (Q_j, A_j), (Q_k, A_k) \rangle$, where $i < j < k \leq t$.

- **Misordered triple sampling**: The three utterance pairs are sampled in a triple as $\langle (Q_i, A_i), (Q_k, A_k), (Q_j, A_j) \rangle$, where $i < j < k \leq t$.

Note that when the current dialogue length $t <= 2$, it is not enough to get a rational sampling for utterance pair triples. Thus, we add three extra shared padding utterance pairs (Q_{-2}, A_{-2}), (Q_{-1}, A_{-1}) and (Q_0, A_0) ahead of all the dialogue data before sampling[1].

Based on above triple sampling strategies, we define the task of inconsistent order detection as: *given a dialogue history* $\{(Q_1, A_1), (Q_2, A_2), \cdots, (Q_t, A_t)\}$ *and the target utterance pair* (Q_t, A_t) *for evaluation, the model needs to predict whether the sampled triple* T *containing* (Q_t, A_t) *is ordered or not*. For instance, $\langle (Q_1, A_1), (Q_2, A_2), (Q_t, A_t) \rangle$ is ordered (output: 0), while $\langle (Q_1, A_1), (Q_t, A_t), (Q_2, A_2) \rangle$ is misordered (output: 1).

[1] Specifically, e.g., for the added padding utterance Q_{-2}, it is represented as a sequence of one same padding word $\{w_1^{(Q_{-2})}, w_2^{(Q_{-2})}, \cdots, w_N^{(Q_{-2})}\}$, where N is the rounded-up averaged length of utterances in the dataset.

3.2 Self-Supervised Network $\mathcal{SSN}$

We plan to build the model to solve the inconsistent order detection task, and explicitly capture the sequential order in dialogue. The overview of our approach is shown in Figure 1. At each dialogue turn t, given a target triple containing the current utterance pair, we first sample triple references from the previous dialogue history to capture more semantic context in dialogue. The target triple and triple references are then transformed into embeddings using an utterance pair encoder and an order reasoning layer. Finally, the concatenation of embeddings is used for the final prediction. We then describe the $\mathcal{SSN}$ in detail as follows.

3.2.1 Triple Reference Sampling

Given the task definition in Section 3.1, the model needs to predict whether there is inconsistent order in the target triple containing the current utterance pair (Q_t, A_t). It is intuitive that if we can get more previous dialogue history, we may make a better prediction for inconsistent order. One trivial way is to encode the full previous dialogue history using a hierarchical network and make the prediction. However, Li et al. (2017) suggests that this structure actually focuses more on the final two preceding utterances instead of the whole history. The sequential order signal is very weak in this condition. We also report some similar results in Section 4.1.

Therefore, we propose a sampling-based approach to model the utterance order based on the dialogue context effectively. For each sampling operation, we sample two triple references T' and T'' from the previous dialogue history $\{(Q_1, A_1), (Q_2, A_2), \cdots, (Q_{t-1}, A_{t-1})\}$ following the sampling strategies in Section 3.1. In general, we explore the following three combinations of reference sampling strategies for T' and T'':

- T' and T'' are sampled ordered references.
- T' and T'' are sampled misordered ones.
- T' is ordered while T'' is misordered.

Note that in our experiments, we choose one certain combination and keep using it for sampling the triple references for all the target triples.

3.2.2 Objective Function

Given the target triple embedding T and the triple reference embedding T' and T'', we use $\mathcal{SSN}$ to calculate the probability $p(T|T', T'') = \mathcal{SSN}(T, T', T'')$. We use the Binary Cross Entropy loss to train the model:

$$L = -\mathbb{E}(y \log p(T|T', T'')), \tag{1}$$

where y is the ground-truth label.

Considering that for the same target triple T, the triple references are sampled m times to approximate the full dialogue history. Then we can rewrite the loss function as

$$L = -\mathbb{E}(\frac{1}{m}\sum_{i=1}^{m} y \log(p^{(i)}(T|T^{(i)'}, T^{(i)''}))), \tag{2}$$

where $T^{(i)'}, T^{(i)''}$ are the triple references of i-th sampling. This is essentially a Monte Carlo estimation and the model would effectively incorporate the dialogue context and capture the order information, avoiding from directly encoding the full dialogue history and the forgetfulness issue.

3.2.3 Network Structure

In this section, we demonstrate how $\mathcal{SSN}$ embeds both the target triple T and triple reference T' and T'' to generate $p(T|T', T'')$ in each sampling.

Utterance Pair Encoder First, given a utterance pair (Q_t, A_t), we concatenate the Q_t and A_t as one sequence. The sequence is then fed into a bidirectional long short-term memory network (LSTM) (Hochreiter and Schmidhuber, 1997), and the utterance pair embedding $\mathbf{U}_t$ is the concatenation of the final two states of the bi-LSTM:

$$\mathbf{U}_t = \begin{bmatrix} \overleftarrow{\mathbf{h}_1} \\ \overrightarrow{\mathbf{h}_{N_t}} \end{bmatrix}, \tag{3}$$

where N_t is the length of the concatenated utterance sequence.

Order Reasoning Layer After obtaining the utterance pair embeddings $(\mathbf{U}_i, \mathbf{U}_j, \mathbf{U}_k)$ of a sampled triple $T =< (Q_i, A_i), (Q_j, A_j), (Q_k, A_k) >$, we need to reason and predict whether there is inconsistent order or not. To simplify our model, we use a 3-step reasoning bi-LSTM with the max-pooling layer to perform the order reasoning:

$$\mathbf{T} = \begin{bmatrix} \text{max-pooling}(\overleftarrow{\mathbf{h}_1}, \overleftarrow{\mathbf{h}_2}, \overleftarrow{\mathbf{h}_3}) \\ \text{max-pooling}(\overrightarrow{\mathbf{h}_1}, \overrightarrow{\mathbf{h}_2}, \overrightarrow{\mathbf{h}_3}) \end{bmatrix}, \tag{4}$$

where the input of each time step in bi-LSTM is one utterance pairs embedding, and $\mathbf{T}$ is the final embedding of the given triple.

Given the target triple embedding $\mathbf{T}$ and the triple reference embedding $\mathbf{T}'$ and $\mathbf{T}''$, the concatenation of these three embeddings is fed into a multi-layer perceptron, returning the probability $p(T|T', T'')$ of the triple is ordered (approaching 0) or misordered (approaching 1).

3.3 Self-Supervised Network for Dialogue

In this section, we explain how the $\mathcal{SSN}$ can be applied to the current dialogue system in both open-domain and task-oriented scenarios.

Suppose we have a dialogue system the the history $\{(Q_1, A_1), \cdots, (Q_{t-1}, A_{t-1})\}$, at turn t, the system generate the utterance A_t based on the Q_t. We can sample a misordered target triple T containing (Q_t, A_t). Following the assumption that the sequential order in a meaningful and coherent dialogue should be unique, the $\mathcal{SSN}$ will be easy to detect the inconsistent order in T if the generated A_t is good. Otherwise, the A_t may be of low quality. Therefore, we take a two-step sampling approach to evaluate the generated utterance A_t using $\mathcal{SSN}$. First, a misordered target triple T containing (Q_t, A_t) is sampled. Then we further sample triple references T' and T'' as in Section 3.2.1 and how easily the misorder in the sampled T can be detected is measured as $\mathbb{E}_{T',T''}(p(T|T', T''))$. Based on the generated utterance A_t, we can sample multiple misordered T, and we set the following expectation to measure the probability that A_t is a good generated utterance:

$$p^*_{\mathcal{SSN}} = \mathbb{E}_{\text{misordered } T}\mathbb{E}_{T',T''}(p(T|T', T'')). \quad (5)$$

In this way, we can view human-generated utterances as good ones, and machine-generated utterances as bad ones. Then we can use the adversarial training methods (Goodfellow et al., 2014; Li et al., 2017; Xu et al., 2017; Su et al., 2018) to train the dialogue system, where $\mathcal{SSN}$ can give clear order-based signal to guide the generator G in the system. The framework of using $\mathcal{SSN}$ with the two-step sampling in real dialogue systems are shown in Figure 2. The objective function then can be formulated as:

$$\begin{aligned} \min_{\theta_G} \max_{\theta_{\mathcal{SSN}}} & \mathbb{E}_{real}[\log p^*_{\mathcal{SSN}}(x)] \\ & + \mathbb{E}_{gen}[\log(1 - p^*_{\mathcal{SSN}}(G(.)))], \end{aligned} \quad (6)$$

where θ_G and $\theta_{\mathcal{SSN}}$ are the parameters of the generator G and SSN in the dialogue systems separately. The x stands for real human-generated utterances, which $G(.)$ represents machine-generated ones. The G and $\mathcal{SSN}$ are alternately updated during training. We further describe the details in open-domain and task-oriented scenarios separately.

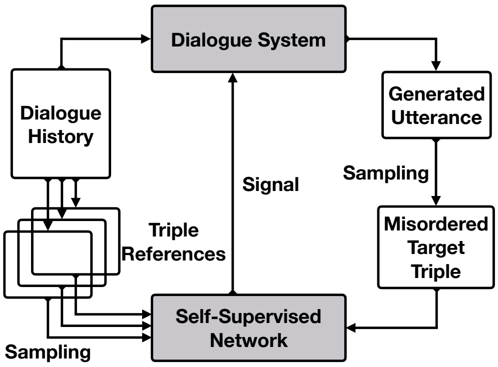


Figure 2: The general framework for dialogue learning with self-supervised network.

3.3.1 Open-Domain Dialogue Learning

The open-domain dialogue task is, given a dialogue history consisting of a sequence of dialogue utterances $\{(Q_1, A_1), \ldots, (Q_{t-1}, A_{t-1})\}$, and current Q_t, the model needs to generate a response utterance A_t. We consider the adversarial training (Li et al., 2017; Xu et al., 2017) for dialogue generation systems. Following the previous approach (Vinyals and Le, 2015; Serban et al., 2016; Luan et al., 2016; Li et al., 2017), we use the SEQ2SEQ model for response generation as the generator G. The SEQ2SEQ first transforms the dialogue history into an embedding using an encoder recurrent network. Conditioned on the history embedding, another decoder recurrent network then computes the probability of tokens at each generation step of the response using a softmax function.

As for the discriminator D, in previous methods, the discriminator directly takes the response utterance A_t with or without the full dialogue history, and predicts whether it is human-generated (output: 1) or machine-generated (output: 0). The probability of being human-generated is set as the reward to update the G using the REINFORCE algorithm (Williams, 1992). As for our $\mathcal{SSN}$, the reward R is set as $R = p^*_{\mathcal{SSN}}$.

3.3.2 Task-Oriented Dialogue Learning

The task-oriented dialogue, usually formulated as a reinforcement learning problem, aims to build a dialogue agent to interact with real users and learn the policy to complete the slot-filling task (Jurafsky and Martin, 2014). While the real-user interaction is expensive and time-consuming, in this scenario, the dialogue systems are often trained with user simulators (Schatzmann et al., 2006; Li et al., 2016c). However, due to the complexity of real conversations and biases in the design of user simulators, the quality of simulated utterances is unstable. Su et al. (2018) propose an adversarial learning approach to differentiate simulated experience from real experience. Following the similar assumption that real-user interactions should be meaningful and coherent, we implement our $\mathcal{SSN}$ instead of the conventional discriminator D to select high-quality stimulated utterances in the task-oriented dialogue systems.

In this scenario, the generator G is the world model which produces simulated user experience, and the $\mathcal{SSN}$ focuses on scoring the simulated user experience Q_t during the training process. Thus, instead of sampling and encoding utterance pairs (Q_t, A_t), here we only use the user utterance Q_t in $\mathcal{SSN}$. We keep other parts of the $\mathcal{SSN}$ remain the same as in Section 3.2. Because the world model G is updated using the multi-task learning without the reward from the $\mathcal{SSN}$, the objective function of the $\mathcal{SSN}$ in Equation 6 can be rewritten as the following during the mini-batch training:

$$\frac{1}{b}\sum_{i=1}^{b}[\log p^{*}_{\mathcal{SSN}}(x^{(i)}) + \log(1 - p^{*}_{\mathcal{SSN}}(G(.)^{(i)}))], \tag{7}$$

where b represents the batch size.

4 Experiments

4.1 Intrinsic Evaluation

Before we deploy the self-supervised network into real dialogue systems, we first test the model architectures for reliability. We randomly choose $40K$ balanced ordered and misordered utterance pair triples from the OpenSubtitles (Tiedemann, 2009) dataset, and train the $\mathcal{SSN}$ to solve this 2-class classification. We sample another $1K$ balanced triples for testing. We also consider a baseline model, where the target triple is encoded by $\mathcal{SSN}$, and the previous dialogue history is encoded by a hierarchical LSTM. The concatenation of two embeddings is used for the final prediction. Because our $\mathcal{SSN}$ is a sampling-based approach, we report the average prediction accuracy of 5 runs on the 2-class classification as shown in Table 1.

Reference Strategy of $\mathcal{SSN}$	**Average Accuracy**
All history by hierarchical LSTM	.694 (.006)
w/o Refers	.670 (.011)
2*Ordered Refers	.740 (.031)
2*misordered Refers	.744 (.029)
1*Ordered + 1*misordered Refers	**.856** (**.017**)

Table 1: The intrinsic evaluation results. The numbers in brackets stand for deviation. Refers: Reference Triples.

From the results, we can observe that: (1) The conventional hierarchical LSTM is not suitable for this task, and this baseline only shows a marginal improvement compared with the strategy that only considers target triple without any history. The results also match previous findings (Li et al., 2017), where they suggest that only the last two proceeding utterances in the hierarchical network are semantically significant. (2) As for our $\mathcal{SSN}$, it is safe to tell that reference triples can be a tremendous supplement to the inconsistent order detection. It is not surprising because by adding reference triples, the $\mathcal{SSN}$ will know more information of semantic context within the dialogue. Especially when having both ordered and misordered references, the $\mathcal{SSN}$ has the highest classification accuracy. This also shows that the sampling strategy, 1*Ordered + 1*misordered references, is the most reliable structure for real dialogue systems. Thus, for the rest of the experiments, we directly use the $\mathcal{SSN}$ with one ordered and one misordered references strategy to achieve the best performance.

4.2 Open-Domain Dialogue Learning

Dataset Following the previous studies (Vinyals and Le, 2015; Li et al., 2017; Xu et al., 2017), we choose the widely-used OpenSubtitles (Tiedemann, 2009) dataset to evaluate different methods. The OpenSubtitles dataset contains movie scripts organized by characters, where we follow Li et al. (2016b) to retain subtitles containing 5-50 words.

Baselines We consider the following two popular adversarial methods for dialogue learning as the baselines:

- **REGS** (Li et al., 2017): The discriminator D takes the full dialogue history by a hierarchi-

Separated G/D	D-REGS	D-AEL	D-$\mathcal{SSN}$
G-REGS	.094	.087	.041
G-AEL	.146	.128	.093
G-$\mathcal{SSN}$	**.203**	**.185**	**.162**

Table 2: The cross evaluation of adversarial success rate on different generators and discriminators. Please refer to Section 4.2 Adversarial Evaluation for explanations.

Model	**distinct-1**	**distinct-2**
REGS	0.0217	0.0695
AEL	0.0311	0.0948
$\mathcal{SSN}$	**0.0393**	**0.1126**

Table 3: The automatic evaluation of generated utterances on distinct-1 and distinct-2 metrics. Please refer to Section 4.2 Automatic Evaluation for explanations.

cal LSTM, and the Monte Carlo search is implemented to obtain rewards for every generation step to update the generator G.

- **AEL** (Xu et al., 2017): The discriminator D only encodes the currently generated utterance by a CNN model and the generator G is optimized using an approximate embedding layer.

Implementation Details We follow the most of parameters in Li et al. (2017); Xu et al. (2017) to make a fair comparison. For the generator model G, we adopt the same SEQ2SEQ model (Sutskever et al., 2014) with an attention mechanism (Bahdanau et al., 2015; Luong et al., 2015) for our approach and baselines. We approximate the dialogue history for G using the concatenation of two preceding utterances following the Li et al. (2017). To train the generator G, we use the REINFORCE algorithm (Williams, 1992) to maximize the expected reward of generated utterances. We also implement the Monte Carlo search to give rewards for each generation step. To accelerate the sampling process, we use multiple GPUs to parallelize and distribute the jobs. As for the $\mathcal{SSN}$, it first gets pre-trained using sampled data from OpenSubtitiles, and then iteratively updated during the min-max adversarial training process. The dimension of the utterance embeddings is 128. The hidden size is 256 for utterance encoding bi-LSTM and 1024 for triple reasoning bi-LSTM. The MLP has a single hidden layer of size 512.

Win	REGS	AEL	$\mathcal{SSN}$
Single-turn Percentage	.095	.192	**.713**
Multi-turn Percentage	.025	.171	**.804**

Table 4: The human evaluation of generated utterances in three methods. The result here is statistically significant with $p < 0.01$ according to sign test. Please refer to Section 4.2 Human Evaluation for explanations.

Adversarial Evaluation Here we use adversarial success rate (AdverSuc), which is the fraction of instances where a G is capable of fooling the D, to evaluate different methods. Higher values of AdverSuc for a dialogue system usually lead to a better response generator. After training three (G, D) using REGS, AEL and $\mathcal{SSN}$, we sample $4K$ dialogue history and use three trained generators to generate response utterances. These machine-generated utterances are then fed into three trained discriminators to see if they are indistinguishable from human-generated ones. The cross evaluation of AdverSuc is shown in Table 2.

From the results, we can observe that: (1) Our trained generator achieve higher AdverSuc in three discriminators, which shows that the generator in our approach can generate more human-like utterance responses. (2) The generators of the other two methods have a noticeable drop in AdverSuc when evaluating on our $\mathcal{SSN}$-based discriminator. This demonstrates that our self-supervised policy for discriminating utterances is successful. (3) The REGS method with full dialogue history encoded performs worse than the AEL that only considers the current utterances. We think this indicates that without explicitly stating the guiding signal, both the generator and the discriminator can be lost about figuring out a good objective function during the training process even when encoding the full history.

Automatic Evaluation For automatic evaluations, we use the two commonly accepted metrics distinct-1 and distinct-2. The distinct-1 and distinct-2, proposed by Li et al. (2016a), are two ways to measure the degree of diversity by calculating the number of distinct unigrams and bigrams in the generated response utterances. The evaluation results are reported in Table 3. The results show that based on the distinct-1 and distinct-2 metrics, the generator trained in our approach can generate relatively more diverse responses. The results are attractive considering that

Agent	Planning Steps	Epoch 100			Epoch 200			Epoch 300		
		Succ	Reward	Turns	Succ	Reward	Turns	Succ	Reward	Turns
D3Q	5	.7467	43.59	14.03	.6800	34.64	15.92	.7200	40.85	13.11
D3Q-$\mathcal{SSN}$		**.7600**	**45.71**	**13.52**	**.7400**	**42.93**	**14.80**	**.7633**	**46.16**	15.24
D3Q (fixed θ_D)		.6800	33.86	17.48	.7000	36.57	16.85	.6933	35.67	17.06
D3Q-$\mathcal{SSN}$ (fixed $\theta_{\mathcal{SSN}}$)		.6633	32.04	16.21	.7133	36.71	17.74	.7067	36.03	**12.91**
D3Q	10	.6333	28.99	16.01	.7000	37.24	**15.52**	.6667	33.09	15.83
D3Q-$\mathcal{SSN}$		**.7800**	**48.71**	15.84	**.8733**	**56.15**	19.57	**.8067**	**50.29**	16.48
D3Q (fixed θ_D)		.7133	36.36	20.48	.8400	54.87	20.48	.7400	42.89	13.81
D3Q-$\mathcal{SSN}$ (fixed $\theta_{\mathcal{SSN}}$)		.7367	42.30	**14.79**	.8300	52.92	18.16	.7933	48.05	**13.73**

Table 5: The experimental results of different dialogue agents at training epoch = $\{100, 200, 300\}$. Each number is averaged over 3 runs, and each run tested on 50 dialogues. The D3Q-$\mathcal{SSN}$ denotes the D3Q agent where our proposed $\mathcal{SSN}$ replaces the discriminator. The "fixed $\theta_D/\theta_{\mathcal{SSN}}$" indicates the discriminator/$\mathcal{SSN}$ is pre-trained and fixed during the training process. Succ: Success Rate. Reward: Average Reward. Turns: Average Turns.

we do not explicitly use a diversity-guided objective function during the training process. We think the reason is that the diverse utterances are easier to reserve the order information. In previous methods, the discriminator D only gives good or bad signals to response generator G, and the G has to figure out what is an acceptable response by itself. As for our $\mathcal{SSN}$, it explicitly forces the G to generate responses that will have unique orders in dialogue, which leads to more diverse utterances.

Human Evaluation For human evaluation, we follow protocols in Li et al. (2016a) and employing crowd-sourced judges from the Amazon Mechanical Turk to evaluate a random sample of 1000 unique generated utterances from three generators in the OpenSubtitles test dataset. We present both the input dialogue history and the generated responses to 5 judges and ask them to decide which one of the three results is the be.ts Ties are not permitted. We consider both single-turn and multi-turn for the evaluation. The results are shown in Table 4. Evidently, the generator trained in our method shows a significant improvement in the quality of generated sentences. The gain is even higher in the multi-turn setting than the single-turn setting. This is because when only considering the single-turn dialogue, the information encoded in three methods will be similar.

4.3 Task-Oriented Dialogue Learning

Dataset Following the previous work (Peng et al., 2018; Su et al., 2018), we use the same Movie-Ticket Booking dataset collected from Amazon Mechanical Turk for evaluation. The dataset is manually labeled based on a schema defined by domain experts consisting of 11 intents and 16 slots in the full domain setting. In total, the dataset has 280 annotated dialogues with an average length of approximately 11 turns. In this scenario, the goal of dialogue systems is to help the user complete the tasks through the conversation.

Baselines We compare our $\mathcal{SSN}$-based discriminator within the state-of-the-art task-oriented dialogue policy learning approach, Discriminative Deep Dyna-Q (D3Q) (Su et al., 2018). At each turn, the D3Q agent takes S planning steps interacting with the simulator and store stimulated user experiences based on the scoring of the discriminator. The stimulated user experiences are generated by the world model, which can be viewed as the generator G in our case. We replace the conventional discriminator D of D3Q with our $\mathcal{SSN}$.

Implementation Details For a fair comparison, we remain most of the parameters in the D3Q algorithm the same as in Su et al. (2018). In the self-supervised network, the dimension of the utterance embeddings is 80. The hidden size is 128 for utterance encoding bi-LSTM and 512 for triple reasoning bi-LSTM. The MLP has a single hidden layer of size 128. We use the simulator[2] as in Li et al. (2016c) to generate user utterances, and the threshold interval is set to a range between 0.45 and 0.55.

Results The experimental results of different agents at training epoch are shown in Table 5. From the results, we can observe that: (1) The D3Q-$\mathcal{SSN}$ outperform the D3Q in the most of cases, which shows that our $\mathcal{SSN}$-based discriminator can improve the ability to recognize

[2] https://github.com/MiuLab/TC-Bot

the high-quality stimulated user experiences. (2) When the planning step increases in D3Q, the performance shows an apparent drop. This is because the discriminator D in the original D3Q agent keeps lots of low-quality stimulated user experiences, which significantly degrade the performance of the D3Q agent. As for our $\mathcal{SSN}$, we can see some performance improvement even when using 10-step planning. This substantially means that our $\mathcal{SSN}$ has a better ability to select the good simulated user experiences, especially in the multi-turn dialogue cases.

5 Conclusion

In this paper, we introduce a self-supervised task, inconsistent order detection, to explicitly capture the order signal of the dialogue. While previous methods suffer from forgetfulness problem when modeling dialogue history, we further propose a sampling-based self-supervised network $\mathcal{SSN}$, to approximately encoding the dialogue history and highlight the order signal. We also show how our $\mathcal{SSN}$ can contribute to real dialogue learning. Empirically, our method advances the previous state-of-the-art dialogue systems in both open-domain and task-oriented scenarios. Theoretically, we believe this self-supervision can be generalized to other types of temporal order in different NLP tasks.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Antoine Bordes, Y-Lan Boureau, and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*.

Andrew M Dai and Quoc V Le. 2015. Semi-supervised sequence learning. In *Proceedings of the 29th Conference on neural information processing systems (NeurIPS)*, pages 3079–3087.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Carl Doersch, Abhinav Gupta, and Alexei A Efros. 2015. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1422–1430.

Jianfeng Gao, Michel Galley, and Lihong Li. 2018. Neural approaches to conversational ai. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2–7.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Proceedings of the 28th Conference on Neural Information Processing Systems (NeurIPS)*, pages 2672–2680.

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

Dan Jurafsky and James H Martin. 2014. *Speech and language processing*. Pearson Education UK.

Alex M Lamb, Anirudh Goyal Alias Parth Goyal, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *Proceedings of the 30th conference on Neural Information Processing Systems (NeurIPS)*, pages 4601–4609.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 110–119.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016b. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1192–1202.

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

Xiujun Li, Zachary C Lipton, Bhuwan Dhingra, Lihong Li, Jianfeng Gao, and Yun-Nung Chen. 2016c. A user simulator for task-completion dialogues. *arXiv preprint arXiv:1612.05688*.

Bing Liu and Ian Lane. 2018. Adversarial learning of task-oriented neural dialog models. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL)*, pages 350–359.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

Yi Luan, Yangfeng Ji, and Mari Ostendorf. 2016. Lstm based conversation models. *arXiv preprint arXiv:1603.09457*.

Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 27th Conference on Neural Information Processing Systems (NeurIPS)*, pages 3111–3119.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. 2016. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2536–2544.

Baolin Peng, Xiujun Li, Jianfeng Gao, Jingjing Liu, and Kam-Fai Wong. 2018. Deep dyna-q: Integrating planning for task-completion dialogue policy learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2182–2192.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 2227–2237.

Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In *Proceedings of the 2011 conference on empirical methods in natural language processing (EMNLP)*, pages 583–593.

Jost Schatzmann, Karl Weilhammer, Matt Stuttle, and Steve Young. 2006. A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The knowledge engineering review*, 21(2):97–126.

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI)*.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*.

Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015a. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM)*, pages 553–562. ACM.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015b. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 196–205.

Shang-Yu Su, Xiujun Li, Jianfeng Gao, Jingjing Liu, and Yun-Nung Chen. 2018. Discriminative deep dyna-q: Robust planning for dialogue policy learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3813–3823.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 28th Conference on Neural Information Processing Systems (NeurIPS)*, pages 3104–3112.

Jörg Tiedemann. 2009. News from opus-a collection of multilingual parallel corpora with tools and interfaces. In *Proceedings of the 2nd Recent advances in natural language processing (RANLP)*, volume 5, pages 237–248.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *ICML Deep Learning Workshop*.

Hong Wang, Xin Wang, Wenhan Xiong, Mo Yu, Xiaoxiao Guo, Shiyu Chang, and William Yang Wang. 2019. Self-supervised learning for contextualized extractive summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Xiaolong Wang and Abhinav Gupta. 2015. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2794–2802.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1711–1721.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gasic, Lina M Rojas Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 438–449.

Jason D Williams, Kavosh Asadi, and Geoffrey Zweig. 2017. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 665–677.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Chen Xing, Yu Wu, Wei Wu, Yalou Huang, and Ming Zhou. 2018. Hierarchical recurrent attention network for response generation. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*.

Zhen Xu, Bingquan Liu, Baoxun Wang, SUN Chengjie, Xiaolong Wang, Zhuoran Wang, and Chao Qi. 2017. Neural response generation via gan with an approximate embedding layer. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 617–626.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2018a. Learning to control the specificity in neural response generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, volume 1, pages 1108–1117.

Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. 2018b. Generating informative and diverse conversational responses via adversarial information maximization. In *Proceedings of the 32nd Conference on Neural Information Processing Systems (NeuIPS)*, pages 1815–1825.